

W0080-01EQ



日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2000年12月20日

出 願 番 号

Application Number:

特願2000-391835

[ST.10/C]:

[JP2000-391835]

出 願 人

Applicant(s):

株式会社日立製作所

RECEIVED

MAR 12 2002

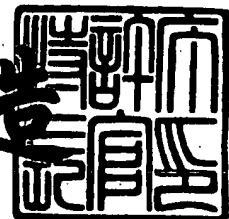
Technology Center 2100

PRIORITY DOCUMENT
CERTIFIED COPY OF

2002年 2月22日

特許庁長官
Commissioner,
Japan Patent Office

及 川 耕 造



【書類名】 特許願

【整理番号】 K00013891

【提出日】 平成12年12月20日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 17/00

【請求項の数】 25

【発明者】

 【住所又は居所】 神奈川県川崎市幸区鹿島田 890 番地 株式会社日立製作所 ビジネスソリューション開発本部内

 【氏名】 紅山 伸夫

【発明者】

 【住所又は居所】 神奈川県川崎市幸区鹿島田 890 番地 株式会社日立製作所 ビジネスソリューション開発本部内

 【氏名】 鈴木 豊人

【発明者】

 【住所又は居所】 神奈川県川崎市幸区鹿島田 890 番地 株式会社日立製作所 ビジネスソリューション開発本部内

 【氏名】 草間 隆人

【特許出願人】

 【識別番号】 000005108

 【氏名又は名称】 株式会社日立製作所

【代理人】

 【識別番号】 100075096

 【弁理士】

 【氏名又は名称】 作田 康夫

【手数料の表示】

 【予納台帳番号】 013088

 【納付金額】 21,000円

【提出物件の目録】

【物件名】	明細書	1
【物件名】	図面	1
【物件名】	要約書	1
【プルーフの要否】	要	

【書類名】 明細書

【発明の名称】 ワークフロー管理方法およびシステム並びにその処理プログラムを格納した記録媒体

【特許請求の範囲】

【請求項1】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理方法において、

前記作業と該作業に対する業務プログラム処理の実行可能数とを対応付けて格納する制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムの数が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値未満であるか判定し、該所定値未満の場合は上記実行する必要がある業務プログラムを実行することを特徴とするワークフロー管理方法。

【請求項2】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理方法において、

前記作業と該作業に対する業務プログラム処理の実行において使用可能な計算機リソース量とを対応付けて格納する制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムと上記実行する必要がある業務プログラムとで使用する計算機リソース量の和が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値以下であるか判定し、所定値以下の場合は上記実行する必要がある業務プログラムを実行することを特徴とするワークフロー管理方法。

【請求項3】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー

一管理方法において、

前記作業と該作業に対する業務プログラム処理の実行可能数とを対応付けて格納する制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要が有る場合、既に実行している業務プログラムの数が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値未満であるか判定し、該所定値未満の場合は上記実行する必要がある業務プログラムの実行を指示する実行要求を出力することを特徴とするワークフロー管理方法。

【請求項4】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理方法において、

前記作業と該作業に対する業務プログラム処理の実行において使用可能な計算機リソース量とを対応付けて格納する制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムと上記実行する必要がある業務プログラムとで使用する計算機リソース量の和が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値以下であるか判定し、所定値以下の場合は上記実行する必要がある業務プログラムの実行を指示する実行要求を出力することを特徴とするワークフロー管理方法。

【請求項5】

請求項1または請求項3記載のいずれかのワークフロー管理方法において、複数の前記作業からなる作業グループを定義し、該作業グループに属する作業に対する業務プログラム処理の実行可能数と該作業グループとを対応付けて格納する制御情報を保持し、

前記判定時には前記実行する必要がある作業を含む上記作業グループを特定し、該作業グループに属する作業において既に実行している業務プログラム数と、該作業グループに対応する該制御情報に格納されている所定値とを比較することを特徴とするワークフロー管理方法。

【請求項 6】

請求項 2 または請求項 4 記載のいずれかのワークフロー管理方法において、複数の前記作業からなる作業グループを定義し、該作業グループに属する作業に対する業務プログラム処理の実行において使用可能な計算機リソース量と該作業グループとを対応付けて格納する制御情報を保持し、前記判定時には前記実行する必要がある作業を含む上記作業グループを特定し、該作業グループに属する作業において既に実行している業務プログラムと上記実行する必要がある業務プログラムとで使用する計算機リソース量の和と、該作業グループに対応する制御情報に格納されている所定値とを比較することを特徴とするワークフロー管理方法。

【請求項 7】

請求項 5 から請求項 6 記載のいずれかのワークフロー管理方法において、前記実行する必要がある作業が複数ある場合、予め作業に設定された優先順に基づいて選択することを特徴とするワークフロー管理方法。

【請求項 8】

請求項 2、請求項 4、請求項 6 または請求項 7 記載のワークフロー管理方法において、前記計算機リソースは、前記実行している業務プログラムが使用するメモリ量であることを特徴とするワークフロー管理方法。

【請求項 9】

請求項 2、請求項 4、請求項 6 または請求項 7 記載のワークフロー管理方法において、前記計算機リソースは、前記実行している業務プログラムが使用するディスク容量であることを特徴とするワークフロー管理方法。

【請求項 10】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理システムにおいて、前記作業と該作業に対する業務プログラム処理の実行可能数とを対応付けて格納

する制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要が有る場合、既に実行している業務プログラムの数が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値未満であるか判定する判定手段と、前記判定手段における判定の結果が所定値未満の場合は上記実行する必要がある業務プログラムを実行する実行手段とを備えたことを特徴とするワークフロー管理システム。

【請求項11】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理システムにおいて、

前記作業と該作業に対する業務プログラム処理の実行において使用可能な計算機リソース量とを対応付けて格納する制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムと上記実行する必要がある業務プログラムとで使用する計算機リソース量の和が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値以下であるか判定する判定手段と、前記判定手段における判定の結果が所定値以下の場合は上記実行する必要がある業務プログラムを実行する実行手段とを備えたことを特徴とするワークフロー管理システム。

【請求項12】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理システムにおいて、

前記作業と該作業に対する業務プログラム処理の実行可能数とを対応付けて格納する制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要が有る場合、既に実行している業務プログラムの数が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値未満であるか判定する判定手段と、前記判定手段における判定の結果が所定値未満の場合は上記実行する必要がある業務プログラムの実行を指示する実行要求を出力する出力手段とを備えたことを特徴とするワークフ

ロー管理システム。

【請求項 13】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理システムにおいて、

前記作業と該作業に対する業務プログラム処理の実行において使用可能な計算機リソース量とを対応付けて格納する制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムと上記実行する必要がある業務プログラムとで使用する計算機リソース量の和が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値以下であるか判定する判定手段と、前記判定手段における判定の結果が所定値以下の場合は上記実行する必要がある業務プログラムの実行を指示する実行要求を出力する出力手段とを備えたことを特徴とするワークフロー管理システム。

【請求項 14】

請求項 10 または請求項 12 記載のいずれかのワークフロー管理システムにおいて、

複数の前記作業からなる作業グループを定義し、該作業グループに属する作業に対する業務プログラム処理の実行可能数と該作業グループとを対応付けて格納する制御情報を保持し、

前記判定時には前記実行する必要がある作業を含む上記作業グループを特定し、該作業グループに属する作業において既に実行している業務プログラム数と、該作業グループに対応する該制御情報に格納されている所定値とを比較することを特徴とするワークフロー管理システム。

【請求項 15】

請求項 11 または請求項 13 記載のいずれかのワークフロー管理システムにおいて、

複数の前記作業からなる作業グループを定義し、該作業グループに属する作業に対する業務プログラム処理の実行において使用可能な計算機リソース量と該作業

グループとを対応付けて格納する制御情報を保持し、
前記判定時には前記実行する必要がある作業を含む上記作業グループを特定し、
該作業グループに属する作業において既に実行している業務プログラムと上記実行する必要のある業務プログラムとで使用する計算機リソース量の和と、該作業グループに対応する制御情報に格納されている所定値とを比較することを特徴とするワークフロー管理システム。

【請求項 1 6】

請求項 1 4 から請求項 1 5 記載のいずれかのワークフロー管理システムにおいて、
前記実行する必要がある作業が複数ある場合、予め作業に設定された優先順に基づいて選択することを特徴とするワークフロー管理システム。

【請求項 1 7】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理プログラムを記録した計算機読み取り可能な記録媒体において、
前記作業と該作業に対する業務プログラム処理の実行可能数とを対応付けて格納する制御情報を保持し、
前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムの数が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値未満であるか判定する判定機能と、前記判定機能における判定の結果が所定値未満の場合は上記実行する必要がある業務プログラムを実行する実行機能とを有するワークフロー管理プログラムを記録したことを特徴とする記録媒体。

【請求項 1 8】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理プログラムを記録した計算機読み取り可能な記録媒体において、
前記作業と該作業に対する業務プログラム処理の実行において使用可能な計算機リソース量とを対応付けて格納する制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムと上記実行する必要がある業務プログラムとで使用する計算機リソース量の和が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値以下であるか判定する判定機能と、前記判定機能における判定の結果が所定値以下の場合は上記実行する必要がある業務プログラムを実行する実行機能とを有するワークフロー管理プログラムを記録したことを特徴とする記録媒体。

【請求項 19】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理プログラムを記録した計算機読み取り可能な記録媒体において、前記作業と該作業に対する業務プログラム処理の実行可能数とを対応付けて格納する制御情報を保持し、前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムの数が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値未満であるか判定する判定機能と、前記判定機能における判定の結果が所定値未満の場合は上記実行する必要がある業務プログラムの実行を指示する実行要求を出力する出力機能とを有するワークフロー管理プログラムを記録したことを特徴とする記録媒体。

【請求項 20】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理プログラムを記録した計算機読み取り可能な記憶媒体において、前記作業と該作業に対する業務プログラム処理の実行において使用可能な計算機リソース量とを対応付けて格納する制御情報を保持し、前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムと上記実行する必要がある業務プログラムとで使用する計算機リソース量の和が前記実行する必要がある作業に対応して前記制御情報に格納されている所定値以下であるか判定する判定機能と、前記判定機能における判定

の結果が所定値以下の場合は上記実行する必要がある業務プログラムの実行を指示する実行要求を出力する出力機能とを有するワークフロー管理プログラムを記録したことを特徴とする記録媒体。

【請求項 2 1】

請求項 1 7 または請求項 1 9 記載のいずれかのワークフロー管理プログラムを記憶した計算機読み取り可能な記憶媒体において、

複数の前記作業からなる作業グループを定義し、該作業グループに属する作業に対する業務プログラム処理の実行可能数と該作業グループとを対応付けて格納する制御情報を保持し、

前記判定時には前記実行する必要がある作業を含む上記作業グループを特定し、該作業グループに属する作業において既に実行している業務プログラム数と、該作業グループに対応する該制御情報に格納されている所定値とを比較することを特徴とするワークフロー管理プログラムを記憶した計算機読み取り可能な記憶媒体。

【請求項 2 2】

請求項 1 8 または請求項 2 0 記載のいずれかのワークフロー管理プログラムを記憶した計算機読み取り可能な記憶媒体において、

複数の前記作業からなる作業グループを定義し、該作業グループに属する作業に対する業務プログラム処理の実行において使用可能な計算機リソース量と該作業グループとを対応付けて格納する制御情報を保持し、

前記判定時には前記実行する必要がある作業を含む上記作業グループを特定し、該作業グループに属する作業において既に実行している業務プログラムと上記実行する必要がある業務プログラムとで使用する計算機リソース量の和と、該作業グループに対応する制御情報に格納されている所定値とを比較することを特徴とするワークフロー管理プログラムを記憶した計算機読み取り可能な記憶媒体。

【請求項 2 3】

請求項 2 1 から請求項 2 2 記載のいずれかのワークフロー管理プログラムを記憶した計算機読み取り可能な記憶媒体において、

前記実行する必要がある作業が複数ある場合、予め作業に設定された優先順に基

づいて選択することを特徴とするワークフロー管理プログラムを記憶した計算機読み取り可能な記憶媒体。

【請求項 2 4】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理方法において、
前記作業と該作業の処理を行う業務プログラムの最大実行数とを対応付けて入力し、該最大実行数に基づいて該作業の処理を行う業務プログラムを実行することにより該実行された業務プログラムの実行数を最大実行数以下とすることを特徴とするワークフロー管理方法。

【請求項 2 5】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理方法において、
前記作業と該作業の処理を行う業務プログラムが使用する最大計算機リソース量とを対応付けて入力し、該最大計算機リソース量に基づいて該作業の処理を行う業務プログラムを実行することにより該実行された業務プログラムが使用する計算機リソース量を最大計算機リソース量以下とすることを特徴とするワークフロー管理方法。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、ワークフロー管理システムの制御技術に関する。

【0 0 0 2】

【従来の技術】

ワークフロー管理システムは、複数の作業からなる業務の流れをプロセス定義として事前定義し、この定義に基づいて作業実行と業務の進行状態の管理を支援する技術である。プロセス定義は、業務の流れである業務フローをモデル化したものと言える。各作業における処理は、人が行う場合や、業務プログラムにより

自動的に実行される場合があるが、ワークフロー管理システムは、これらの作業の人への割当てや業務プログラムによる処理の実行制御を行い、且つ業務フロー全体としての状態管理、進捗管理を行うものである。ワークフロー管理システムは、主にホワイトカラーの生産性向上を目的とし、オフィス業務の手続きをコンピュータを用いて自動化すると言った観点から発展してきた (Workflow Management Coalition, The Workflow Reference Model, TC00-1003, January 1995, Chap. 2)。

【 0 0 0 3 】

こうしたワークフロー管理システムは、主にグループウェアとして比較的小規模なシステム形態で利用され始めたが、徐々に基幹系業務でも利用され、より大規模で高スループットなシステムへの適用が行われている。こうした中で、例えば特開平 1 1 - 2 1 3 0 8 2 号公報「ワークフロー管理装置及びワークフロー管理システム並びこれらのプログラムを記憶したコンピュータ読み取り可能な記憶媒体」では、大規模なワークフロー管理システムにおいて効率よくワークフロー管理を行う方法について述べている。

【 0 0 0 4 】

ワークフロー管理システムにおいて、ワークフロー実行制御エンジンはプロセス定義に従ってワークフローの実行管理を行う。一般に、ワークフロー管理システムでは、複数の案件に関する業務フローが同時に進行する為、大量の案件が同時に投入されると、ワークフロー実行制御エンジンによるワークフロー実行管理の負荷が増大する。また他方で、ある時点において実行可能な作業数も増加する為、作業が人手を介さずに業務プログラムで処理される場合、業務プログラム処理の実行計画 (スケジューリング) や負荷制御を行うことが必要となる。

【 0 0 0 5 】

上記の発明は、複数のワークフロー実行制御エンジンにワークフロー実行管理の負荷を分散させる事で、ワークフロー実行管理の負荷が増大するという課題を解決する技術について述べている。

【 0 0 0 6 】

【発明が解決しようとする課題】

従来技術では、プロセス定義とは独立に業務プログラム側で計算機リソースの配分を考慮した負荷制御や処理スケジューリングを行う必要があるため、業務フローのモデルに応じて適切に業務プログラム処理をスケジューリングし負荷制御を行う事が困難であった。

【 0 0 0 7 】

本発明の目的は、作業に対する処理を実行する業務プログラムが使用する計算機リソースの配分を考慮したワークフロー管理方法およびシステムを提供することにある。

【 0 0 0 8 】

【課題を解決するための手段】

本発明では、前記課題を改善するために、前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムの数または計算機リソース量が前記実行する必要がある作業に対応して格納されている所定値以下であるか判定し、該所定値以下の場合は上記実行する必要がある業務プログラムを実行する。

【 0 0 0 9 】

このようにすることにより、作業に対する処理を実行する業務プログラムが使用する計算機リソースの配分を考慮したワークフロー管理方法およびシステムの提供が可能となる。

【 0 0 1 0 】

【発明の実施の形態】

以下、図を用いて本発明の実施の形態の一実施例を示す。

【 0 0 1 1 】

以下の実施例におけるワークフロー管理システムの例は、本発明の説明に不要な機能や詳細は省略している為、一般的なワークフロー管理システムに比べて単純化されているが、本発明の適用範囲が制限されるものではない。

【0012】

図1は、本発明におけるワークフロー管理システムの構成の一例である。

計算機0131は、表示装置0151、CPU0152、通信装置0153、補助記憶装置0154、メモリ0155、キーボード0156、マウス0157、の各ハードウェアで構成される。図中では簡単の為、一部の計算機0131に関してのみ計算機内部の構成の詳細を記しているが、他の計算機0131に対しても同様のハードウェア構成でよい。ワークフロー実行制御エンジン0101、クライアント業務プログラム0102、自動実行業務プログラム0103はソフトウェアプログラムであり、計算機0131上で動作する。これらのソフトウェアプログラムは補助記憶装置0154に格納されているが、実行時にメモリ0155に読込まれ、CPU0152により実行される。また、これらのソフトウェアプログラムは、表示装置0151や、キーボード0156、マウス0157等の入力装置を用いて、人である作業者と対話を行ったり、通信装置0203を用いて他の計算機上で動作するソフトウェアプログラムと互いに通信する事が可能である。また、これらの各プログラムは、同一の計算機0131上に配置しても、コンピュータネットワーク0104で相互に接続された異なる計算機0131上に配置しても構わない。

【0013】

ワークフロー実行制御エンジン0101は、事前定義された作業の処理手順を記述したプロセス定義0121に基づいて、作業や業務フローの実行状態管理を行う。また、ワークフロー実行制御エンジン0101は、コンピュータネットワーク0104を介してクライアント業務プログラム0102からの処理要求を受け付け、自動実行業務プログラム0103に対して処理要求を送信する。ワークフロー実行制御エンジン0101の動作については、後述する。

【0014】

クライアント業務プログラム0102は、人である作業者が直接操作する事で、その作業者に割当てられた作業のリストをワークフロー制御エンジン0101から取得し表示したり、割当てられた作業に対する業務処理を行う。また、作業に対する業務処理の完了後、ワークフロー実行制御エンジン0101へ作業の完

了通知を行う。

【0015】

自動実行業務プログラム0103は、ワークフロー実行制御エンジン0101より業務処理の実行要求を受付け、作業に対する業務処理を人手を介さずに実行し、業務処理の完了後に完了通知を返す。

【0016】

事前定義されたプロセス定義0121は、補助記憶装置0154に複数格納出来る。図2に、プロセス定義0121の定義イメージを示す。プロセス定義0121は、複数の作業定義0202をアロー0201で連結した有効グラフとして表現される。グラフの開始点や終了点、分岐点や合流点は、制御ノードと呼ぶ有効グラフ上の特殊なノードを用いて表現する。図2中のノード0203～0206が制御ノードであり、それぞれ開始ノード0203、終了ノード0204、分岐ノード0205、合流ノード0206である。有効グラフは、開始ノード0203で始まり、終了ノード0204で終了する。図2に示すプロセス定義0121の詳細と、補助記憶装置0154への格納形式を図3～図5に示す。図3はプロセス定義0121内に含まれる作業定義0202の定義情報を格納する作業定義情報テーブル0311、図4はプロセス定義0121内に含まれる制御ノード0203～0206の定義情報を格納する制御ノード定義情報テーブル0411、図5は作業定義0211及び制御ノード0203～0206間を接続するアロー0201の定義情報を格納するアロー定義情報テーブル0511の詳細を示す図である。何れのテーブルも複数のプロセス定義0121に関する情報を格納する為、どのプロセス定義0121に属する定義情報かを特定する為のプロセス定義名を第1カラムに持つ（図中のカラム、0301、0401、0501）。以下、各図毎にカラムの詳細を説明する。

【0017】

図3において、作業定義名0302は作業定義0202を一意に特定する為の名称である。作業種別0303は、「作業者実行」及び「自動実行」の2種類の値をとり、該作業定義における作業の処理を、人である作業者がクライアント業務プログラム0102を操作することで行うか（「作業者実行」）、ワークフロ

ー実行制御エンジン0101が自動実行業務プログラム0103へ処理要求を行うことで人手を介さずに実行するか（「自動実行」）、を指定する。作業種別0304は、作業種別0303が「作業種別0303」の場合に、作業を処理すべき作業種別を特定する名前を指定する。業務プログラム情報0305は、作業種別0303が「自動実行」の場合に、作業を処理すべき自動実行業務プログラム0103を特定する為の情報を指定する。

【0018】

図4において、制御ノード名0402は制御ノード0203～0206の定義情報を一意に特定する為の名称である。ノード種別0403は、「開始」、「終了」、「分岐」、「合流」の4種類の値を取り、該制御ノードの種類を示す。

【0019】

図5において、遷移元0502及び遷移先0503は、2つの作業定義0202または制御ノード0203～0206間をアロー0201で接続した際に、接続されたノードの作業定義名0302もしくは制御ノード名0402を指定する。

【0020】

ワークフロー実行状態情報0122は、ワークフロー実行制御エンジン0101による作業や業務フローの実行状態管理において、それらの実行状態を案件単位で補助記憶装置0154に格納するものである。ワークフロー実行状態情報0122の格納形式の詳細を図6に示す。図6に示す状態管理テーブル0611の各レコードは、案件内の各作業の状態を保持している。案件名0601は、各案件を一意に特定する名称である。プロセス定義名0602は、該案件の雛型であるプロセス定義0121を示す。作業定義名0603は、状態を保持している作業の雛型である作業定義0202を特定する。状態0604は、作業の実行状態を表し、「Initial」、「Ready」、「Completed」の3種類の値をとる。「Initial」状態は、作業の初期状態であり、案件開始時点で全作業の状態はこの状態である。「Ready」状態は、作業に対する処理が実行開始可能である事を示す状態である。「Completed」状態は、作業に対する処理の完了を示す状態である。作業種別0605は、作業種別が「作業

者実行」の場合の、作業を実行すべき作業者の名前である。

【0021】

次に、ワークフロー実行制御エンジン0101の動作について説明する。

【0022】

図1に示す通り、ワークフロー実行制御エンジン0101は、ワークフロー実行状態制御部0111、業務プログラム実行制御部0112、クライアント要求処理部0113、の各サブプログラムから構成される。

【0023】

クライアント要求処理部0113は、作業リスト取得コマンド「GetWorkList」、作業完了通知コマンド「CompleteWork」、及び案件開始コマンド「StartProcess」の3種類のコマンドを、クライアント業務プログラム0102より受け付ける。「GetWorkList」コマンドは、作業者名を入力として受取り、その作業者に割当てられた作業の作業名リストを出力する。「CompleteWork」コマンドは、案件名、及び作業定義名を入力として受取り、指定された案件の指定された作業定義名に対する作業の処理完了をワークフロー実行状態制御部0111へ通知する。「StartProcess」コマンドは、案件名及びプロセス定義名を入力として受取り、指定されたプロセス定義の案件を指定された案件名で開始する事をワークフロー実行状態制御部0111へ依頼する。

【0024】

ワークフロー実行状態制御部0111は、クライアント要求処理部0113の受付けた「StartProcess」コマンド、及び「CompleteWork」コマンドに応じて、プロセス定義0121に従いワークフロー実行状態0122を適切に変更する。「StartProcess」コマンド受付け時には、指定されたプロセス定義0121において定義された各作業定義0202の情報を作業定義情報テーブル0311からプロセス定義名をキーに検索し、各作業定義0202に対応する作業のレコードを状態管理テーブル0611へ挿入する。この時、各作業の状態0604は「Initial」状態としておく。また、案件名0601及びプロセス定義名0602の各カラムには、夫々コマンド受付

け時に指定された案件名とプロセス定義名を設定し、作業定義名0603及び作業0605の各カラムには該レコードに対応する作業定義0202の属性情報である作業定義名0302及び作業0304を設定する。更に、プロセス定義0121の有効グラフに従って開始ノード0203から辿って最初に実行すべき作業定義0202を決定し、該作業定義に対応する作業の状態0605を「Ready」状態とする。この時、該作業定義0202の作業種別0303が「自動実行」の場合、業務プログラム実行制御部0112へ自動実行業務プログラムによる業務処理の実行を依頼する。「CompleteWork」コマンド受け付け時には、指定された案件の指定された作業定義に対する作業の状態を「Completed」状態にし、プロセス定義0121の有効グラフに従って該作業定義の次に実行すべき作業定義を決定し、次に実行すべき作業定義に対する作業の状態0605を「Ready」状態とする。この時、作業種別0303が「自動実行」の場合、前述したように業務プログラム実行制御部0112へ自動実行業務プログラムによる業務処理の実行を依頼する。

【0025】

業務プログラム実行制御部0112は、ワークフロー実行状態制御部0111より依頼された自動実行業務プログラムの実行要求に従い、自動実行業務プログラムへ業務処理の実行要求を送信し、業務処理完了後に完了通知を受取る。更に、業務プログラム実行制御部0112はワークフロー実行状態制御部0111へ業務処理の完了通知を行う。この際、ワークフロー実行状態制御部は、「CompleteWork」コマンド受け付け時と同様の処理を行う。これにより、プロセス定義0121において次に処理すべき作業へ業務フローを遷移させることが出来る。

【0026】

以上で、本発明の一実施例におけるワークフロー管理システムの構成について説明した。

【0027】

次に、上述したワークフロー管理システムにおいて、本発明を特徴付ける機能を付加えた業務プログラム実行制御部0112の動作の詳細について説明する。

【 0 0 2 8 】

図 7 に示す様に、本発明における業務プログラム実行制御部 0 1 1 2 は、更に業務プログラム実行要求管理部 0 7 0 1、業務プログラム実行計画部 0 7 0 2、業務プログラム起動制御部 0 7 0 3 の各サブプログラムから構成される。業務プログラム実行要求管理部 0 7 0 1 は、ワークフロー実行状態制御部 0 1 1 1 から受取った自動実行業務プログラム 0 1 0 3 への業務処理の実行要求を補助記憶装置 0 1 5 4 内に一旦格納し（図 7 の業務プログラム実行要求 0 7 1 1）、業務プログラム実行計画部 0 7 0 2 の実行計画に従い適切なタイミングで処理されるまで管理する。業務プログラム実行計画部 0 7 0 2 は、自動実行業務プログラム 0 1 0 3 による業務処理の実行計画方法を事前に定義した業務プログラム実行スケジュール定義 0 7 1 2 に基づいて、業務プログラム実行要求 0 7 1 1 の実行計画を行い、実行を決定した自動実行業務プログラムの実行要求を業務プログラム起動制御部 0 7 0 3 へ送る。また、自動実行業務プログラム 0 1 0 3 による業務処理完了時に、完了通知を業務プログラム起動制御部 0 7 0 3 より受取る。業務プログラム起動制御部 0 7 0 3 は、自動実行業務プログラム 0 1 0 3 へ業務処理の実行要求を送信し、自動実行業務プログラム 0 1 0 3 による処理完了後、完了通知を受信する。

【 0 0 2 9 】

図 8 に、本実施例における業務プログラム実行スケジュール定義 0 7 1 2 の定義方法と具体的な定義例を示す。業務プログラム実行スケジュール定義 0 7 1 2 は、ワークフロー管理システムにおいて定義され登録されている 1 つ以上のプロセス定義 0 1 2 1 に渡って記述される。先ず、スケジューリング制御単位 0 8 0 1 と呼ぶ 1 つ以上の作業定義 0 2 0 2 の集合を定義する。この時、スケジューリング制御単位 0 8 0 1 に含める作業定義 0 2 0 2 の作業種別は、「自動実行」のものに限る。次に各スケジューリング制御単位 0 8 0 1 毎にスケジューリング制御単位 0 8 0 1 内での実行計画方法を指定する。スケジューリング制御単位 0 8 0 1 内での実行計画方法の指定には、最大実行多重度（スケジューリング制御単位 0 8 0 1 毎に指定）と作業定義間での処理の優先度（作業定義 0 2 0 2 毎に指定）の 2 通りの方法がある。優先度としては、「高」、「中」、「低」の 3 レベ

ルを指定可能である。最大実行多重度は、該スケジューリング制御単位0801において同時に実行可能な自動実行業務プログラム0103による業務処理の上限を示す設定値である。優先度は、同一スケジューリング制御単位0801内において、自動実行業務プログラム0103による業務処理の実行順序の優先順位を、作業定義0202毎に指定するものである。同一スケジューリング制御単位0801内では、より高い優先度を指定した作業定義における自動実行業務プログラムの業務処理実行要求が、先に実行される。図8中では、優先度は各作業定義0202の上側に、同様に実行多重度は各スケジューリング制御単位の上側に記述した。

【0030】

図9、図10に、業務プログラム実行スケジュール定義0712の補助記憶装置0154への格納形式と、図8に示した定義例の格納状態を示す。図9は業務プログラム実行スケジュール定義0712に含まれるスケジューリング制御単位0801の定義情報を格納するスケジューリング制御単位定義情報テーブル0911、図10は各スケジューリング制御単位0801に含まれる作業種別が「自動実行」の作業定義0202に対する定義情報を格納するスケジューリング制御単位作業定義情報テーブル1011を示す図である。

【0031】

図9において、スケジューリング制御単位名0901はスケジューリング制御単位0801を一意に特定する為の名称である。最大実行多重度0902は、該スケジューリング制御単位0801に対する最大実行多重度の指定値である。

【0032】

図10において、プロセス定義名1002及び作業定義名1003は、スケジューリング制御単位0801に含まれる作業定義0202を特定する為のものである。スケジューリング制御単位名1001は、プロセス定義1002及び作業定義名1003にて特定された作業定義0202の属するスケジューリング制御単位0801を特定する為のものである。優先度1004は、該作業定義0202の該スケジューリング制御単位0801内における優先度の指定値である。

【0033】

図11に、業務プログラム実行要求0711の補助記憶装置0154への格納形式を示す。業務プログラム実行要求0711は、業務プログラム実行要求管理テーブル1111内に、各実行要求毎に1レコードずつ格納される。プロセス定義名1101、作業定義名1102、及び案件名1103は、実行要求の発生元である作業（「自動実行」）を一意に特定するものである。業務プログラム情報1104は、該実行要求に対する業務処理を行う自動実行業務プログラム0103を特定する為の情報であり、作業定義情報テーブル0311の業務プログラム情報0305カラムから取得される。発生日時1105は、実行要求の発生した日時である。

【0034】

以下、業務プログラム実行制御部0112の各部の動作の詳細を順に示す。

【0035】

まず、業務プログラム実行要求管理部0701の動作の詳細を説明する。業務プログラム実行要求管理部0701は、ワークフロー実行状態制御部0111により呼出される「実行要求登録処理」と、業務プログラム実行計画部0702により呼出される「実行要求取得処理」の2通りの処理を実行する。ワークフロー実行状態制御部0111は、作業種別が「自動実行」の作業が新たに「Ready」状態になった場合に、「実行要求登録処理」を呼出して自動実行業務プログラム0103による業務処理の実行要求を業務プログラム実行要求管理部0701へ登録する。この際、ワークフロー実行状態制御部0111は、状態管理テーブル0611及び作業定義情報テーブル0311から、業務処理の実行要求を行う作業に関する情報（案件名0601、プロセス定義名0602、作業定義名0603、業務プログラム情報0305）を取得し、業務プログラム実行要求管理部0701へ受渡す。業務プログラム情報0305は、プロセス定義名0301及び作業定義名0302をキーに、作業定義情報テーブル0311から該作業に関する作業定義情報を検索する事で取得できる。

【0036】

以下、「実行要求登録処理」の処理手順を順を追って説明する。まず、業務プ

プログラム実行要求管理部0701は、ワークフロー実行状態制御部0111より自動実行業務プログラム0103の実行要求を受取る。この際、実行要求の属性情報として、プロセス定義名、作業定義名、案件名、及び業務プログラム情報を受取る事が出来る。次に、現在の日時を取得し、ワークフロー実行状態制御部0111より受取った実行要求を、取得した現在日時と併せて業務プログラム実行要求管理テーブル1111へ挿入する。現在日時は、例えばオペレーティングシステムの提供するシステムコールを呼出すことで取得することが出来る。最後に、業務プログラム実行計画部0702へ、新たに実行要求が発生したことを通知する。これにより、業務プログラム実行計画部0702の実行計画処理起動トリガを起すことが出来る。実行計画処理起動トリガについては、後述する。

【0037】

次に、図12のフローチャートを用いて、「実行要求取得処理」の処理手順を示す。(ステップ1201)業務プログラム実行計画部0702は、作業定義(プロセス定義名、及び作業定義名の組)のリストを指定して、「実行要求取得処理」を呼出す。(ステップ1202)ステップ1201で渡された作業定義リストに属する実行要求を、業務プログラム実行要求管理テーブル1111より検索し、発生日時の古い順にソートする。(ステップ1203)条件に合う実行要求が存在しない場合、処理を終了する。条件に合う実行要求が存在する場合、ステップ1204へ進む。(ステップ1204)ステップ1202で検索した実行要求の内、最も古い実行要求のレコードを読み込み、業務プログラム実行要求管理テーブル1111から削除する。(ステップ1205)ステップ1204で取得した実行要求を、業務プログラム実行計画部0702へ返す。

【0038】

次に、業務プログラム実行計画部0702の動作の詳細を説明する。業務プログラム実行計画部0702は、「初期処理」、「実行計画処理」、「完了通知受取り処理」の3通りの処理を実行する。以下、順を追って各処理の詳細を説明する。

【0039】

最初に、ワークフロー実行制御エンジン0101の起動時に一度のみ実行され

る、業務プログラム実行計画部0702の「初期処理」の処理手順を示す。まず、業務プログラム実行スケジュール定義0712内のスケジューリング制御単位定義情報テーブル0911を検索し、スケジューリング制御単位のリストを取得する。次に、各スケジューリング制御単位毎に、自動実行業務プログラム0103による業務処理の現時点での実行数を保持する変数領域（実行数カウンタ）を確保し、各実行数カウンタを0に初期化する。最後に、業務プログラム実行計画部0702の実行計画処理起動トリガを起す。以上で、「初期処理」の処理手順は完了する。

【0040】

業務プログラム実行計画部0702は、実行計画処理起動トリガの発生を受けて、自動実行業務プログラム0103による業務処理の実行要求に対する「実行計画処理」を開始する。「実行計画処理」では、業務プログラム実行スケジュール定義0712に基づいて、補助記憶装置0154に格納された業務プログラム実行要求0711の実行計画を行い、実行開始可能な実行要求を決定する。更に、実行開始可能と決定した実行要求に対する自動実行業務プログラム0103による業務処理実行を、業務プログラム起動制御部0703へ依頼する。図13～図14に、「実行計画処理」の詳細な処理手順を示す。

【0041】

図13に示す処理手順では、各スケジューリング制御単位毎に現在の実行数カウンタの値と最大実行多重度とを比較し、同時実行数に余裕のあるスケジューリング制御単位を決定する。以下、各ステップを追って詳細に説明する。（ステップ1301）業務プログラム実行スケジュール定義0712から、スケジューリング制御単位のリストを取得し、最大実行多重度の設定値を得る。具体的には、スケジューリング制御単位定義情報テーブル0911を検索し、スケジューリング制御単位名0901と対応する最大実行多重度0902の組のリストを取得する。（ステップ1302）未チェックのスケジューリング制御単位を1つ選択し、現在の実行数カウンタ値と最大実行多重度の設定値とを比較する。（ステップ1303）ステップ1302での比較の結果、同時実行数に余裕が無い場合、ステップ1305へ進む。同時実行数に余裕が有る場合、ステップ1304へ進む

。(ステップ1304) 当該スケジューリング制御単位における自動実行業務プログラムの実行要求を、最大実行多重度の制限範囲内で実行開始する。このステップの処理は、図14のフローチャートにおいて後ほど詳しく説明する。(ステップ1305) ステップ1302にて選択したスケジューリング制御単位を、チェック済みにマークする。(ステップ1306) 未チェックのスケジューリング制御単位が残っている場合、ステップ1302へ戻り、次のスケジューリング制御単位を選択して同様の処理を繰り返す。全てのスケジューリング制御単位に対する処理が完了したら、「実行計画処理」を終了する。

【0042】

図14に示す処理では、図13の処理で決定した同時実行数に余裕のあるスケジューリング制御単位において、自動実行業務プログラム0103の業務処理に対する実行要求の実行順序を決定する。図14の処理フローは、図13のステップ1304の処理を詳細に示したものである。以下、各ステップ毎に説明する。

(ステップ1401) 未処理の優先度レベルを、優先度の高い順に1つ選択する。最初は、優先度「高」から開始する。(ステップ1402) 図13のステップ1304において選択されているスケジューリング制御単位に属し、且つステップ1401で選択した優先度レベルの作業定義のリストを、スケジューリング制御単位作業定義情報テーブル1011より検索する。(ステップ1403) ステップ1402で検索した作業定義のリストを指定して、業務プログラム実行要求管理部0701の「実行要求取得処理」を呼出す。これにより、指定した作業定義に属する実行要求の内、発生日時の最も古いものを1つ取得できる。(ステップ1405) ステップ1403にて実行要求が取得できた場合、取得した実行要求に対する自動実行業務プログラム0103の業務処理の実行を、業務プログラム起動制御部0703へ依頼する。この時、業務プログラム起動制御部0703へは、取得した実行要求の各属性値(プロセス定義名1101、作業定義名1102、案件名1103、及び業務プログラム情報1104)をパラメタとして受渡す。(ステップ1406) 当該スケジューリング制御単位の現在の実行数カウンタを1だけ加算し、当該スケジューリング制御単位における最大実行多重度と比較する。(ステップ1407) ステップ1406での比較において同時実行数

に余裕がある場合、ステップ1403へ戻り、当該スケジューリング制御単位の当該優先度における他の実行要求に対する処理を繰り返す。（ステップ1408）ステップ1403において、選択した優先度の実行要求が存在しなかった場合、当該優先度を処理済にマークする。（ステップ1409）未処理の優先度が存在する場合、ステップ1401へ戻り、次に高い優先度レベルを選択し直して上記処理を繰り返す。

【0043】

最後に、「完了通知受取り処理」の処理手順を示す。「完了通知受取り処理」は、業務プログラム起動制御部0703から、自動実行業務プログラム0103による業務処理の完了通知を受取る際に実行される。この完了通知は、「実行計画処理」（図14ステップ1405）において業務プログラム起動制御部0703へ依頼した、自動実行業務プログラム0103による業務処理の実行依頼に対するものである。処理手順は、以下の様になる。まず、業務プログラム起動制御部0703から、業務処理の完了通知を受取る。この時、業務処理実行の依頼時に受け渡した実行要求のパラメタの内、作業を特定する為の情報（プロセス定義名、作業定義名、案件名）を受取る。次に、完了通知を受取った実行要求に対する作業定義0202の属するスケジューリング制御単位0801を、スケジューリング制御単位作業定義情報テーブル1011より検索し、特定する。更に、特定したスケジューリング制御単位0801における、現在の実行数カウンタを1だけ減算する。次に、ワークフロー実行状態制御部0111へ、当該実行要求に対する作業の業務処理の完了を通知する。最後に、業務プログラム実行計画部0702の実行計画処理起動トリガを起す。

【0044】

実行計画処理起動トリガは、業務プログラム実行計画部0702において実行計画を再実行する必要がある場合に発生させるもので、以下の3通りのケースがある。第1のケースは、新たに実行要求が発生した場合で、業務プログラム実行要求管理部0701が、ワークフロー実行状態制御部0111より自動実行業務プログラム0103の実行要求を受取った際に発生する。業務プログラム実行要求管理部0701の「実行要求登録処理」内での前記トリガの発生が、このケー

スにあたる。第2のケースは、自動実行業務プログラム0103による業務処理の完了により同時実行数に余裕が発生する場合である。業務プログラム実行計画部0702の「完了通知受取り処理」内での前記トリガの発生が、このケースにあたる。第3のケースは、ワークフロー実行制御エンジン0101の起動時であり、この場合、自動実行業務プログラム0103による業務処理は何れもまだ行われておらず、同時実行数に余裕がある。業務プログラム実行計画部0702の「初期処理」内での前記トリガの発生が、このケースにあたる。

【0045】

最後に、業務プログラム起動制御部0703について述べる。業務プログラム起動制御部0703は、自動実行業務プログラム0103へ業務処理の実行要求を送信し、自動実行業務プログラム0103による業務処理の完了後、完了通知を受信する。本実施例においては、業務プログラム起動制御部0703と自動実行業務プログラム0103との通信は、通信装置0153によりコンピュータネットワーク0104を介して行う。ただし、他の実施の形態においては、様々な通信手段を採り得る。例えば、業務プログラム起動制御部0703と自動実行業務プログラム0103とが同一計算機0131内で動作する場合、パイプやシグナル、共用メモリ等のオペレーティングシステムの提供するプロセス間通信手段を利用しても良い。この場合、自動実行業務プログラム0103のプロセスの開始や終了を、業務プログラム起動制御部0703で制御することも可能である。

【0046】

以上で、業務プログラム実行制御部0112を構成する各部の動作について説明した。

【0047】

次に、業務プログラム実行制御部0112の動作の理解を助ける目的で、図8～図10に示すプロセス定義、及び業務プログラム実行スケジュール定義の定義例と、図11に示す業務プログラム実行要求の格納状態の例の組合せにおいて、どの様に実行計画処理が行われるかを具体的を示す。

【0048】

図8～図10に示す業務プログラムスケジュール定義の定義例では、2つのプ

プロセス定義（「PD2」、及び「PD3」）上に、3つのスケジューリング制御単位（「SC1」、「SC2」、及び「SC3」）を定義している。各スケジューリング制御単位にはそれぞれ2つずつの作業定義が属しており、「SC1」には「PD2:A2」「PD2:B2」、「SC2」には「PD3:A3」「PD3:B3」、「SC3」には「PD2:C2」「PD3:C3」がそれぞれ含まれる（ここで、作業定義を指定する際の記法として、作業定義名の前にコロン「:」で区切ってプロセス定義名を付与して記述する。また、プロセス定義名が文脈から明らかな場合は、省略することもある。）。各スケジューリング定義単位には、最大実行多重度として、「SC1」には2、「SC2」には3、「SC3」には5、が指定されている。業務プログラム実行制御部0112は、各スケジューリング制御単位0801内の全作業定義0202における自動実行業務プログラム0103の業務処理の同時実行数が、指定された最大実行多重度を上回らない様に制御する。これらの作業定義の内、スケジュール制御単位「SC3」に属する2つの作業定義「PD2:C2」及び「PD3:C3」には、それぞれ優先度として「高」及び「低」が割付けられている。またその他の作業定義に対しての優先度は、全て「中」である。優先度の指定は、各スケジューリング制御単位0801内における自動実行業務プログラム0103による業務処理の実行順序を、各作業定義に対して指定した優先度順に行うように、業務プログラム実行制御部0112へ指示するものである。この定義例では、スケジューリング制御単位「SC3」において、作業定義「PD2:C2」における業務処理が作業定義「PD3:C3」における業務処理より優先的に実行される。その他のスケジューリング制御単位「SC1」及び「SC2」においては、全ての作業定義の優先度が等しい為、作業定義による違いは無く平等に発生日時順に処理される。ここで、これらの作業定義は全て作業種別が「自動実行」であるとし、作業定義「PD2:A2」「PD3:A3」は共に業務プログラム「ProgramA」を、作業定義「PD2:B2」「PD3:B3」は共に業務プログラム「ProgramB」を、作業定義「PD2:C2」「PD3:C3」は共に業務プログラム「ProgramC」を用いて、業務処理を行うものとする。即ち、作業情報定義テーブル0311の業務プログラム情報0305のカラムに、各作業定義に

対応して業務プログラム名が記されているものとする。

【0049】

この業務プログラムスケジュール定義の定義例により、業務プログラム実行制御部0112は自動実行業務プログラム0103による業務処理の実行計画処理を、以下の様に行う。ここで、実行計画処理時に、各スケジューリング制御単位における実行数カウントは、「SC1」及び「SC2」においては共に2、「SC3」においては4であるとする。即ち、同時実行数に余裕があるのは、「SC2」及び「SC3」においてであり、余裕はそれぞれ1多重ずつである。説明には、再び業務プログラム実行計画部0702のフローチャートである図13～図14を用いる。

【0050】

まず、ステップ1301において、スケジューリング制御単位定義情報テーブル0911を検索し、スケジューリング制御単位のリストとして、「SC1」（最大実行多重度2）、「SC2」（最大実行多重度3）及び「SC3」（最大実行多重度5）を取得する。次に、ステップ1302において、スケジューリング制御単位「SC1」を選択し、現在の実行数カウント2と最大実行多重度2とを比較する。この場合、実行数カウントに余裕が無い為（ステップ1303）、スケジューリング制御単位「SC1」をチェック済みにマークする（ステップ1305）。この時点で、まだ未チェックのスケジューリング制御単位「SC2」及び「SC3」が残っている為（ステップ1306）、ステップ1302にて今度は「SC2」を選択する。「SC2」の最大実行多重度は3で、現在の実行数カウント2と比較して同時実行数に1だけ余裕がある為、ステップ1304にてスケジューリング制御単位「SC2」の実行要求に対する業務処理を開始する。ステップ1304における処理は、図14に示すフローチャートの通り行われるが、具体的には以下の様になる。

【0051】

先ず、ステップ1401にて優先度レベル「高」を選択し、ステップ1402にてスケジューリング制御単位「SC2」に属する優先度レベル「高」の作業定義リストを検索する。該当する作業定義は存在しない為、ステップ1403にお

いても該当する実行要求は存在せず、ステップ1408にて優先度レベル「高」を処理済にマークする。次に、再びステップ1402において優先度レベル「中」を選択し、今度はステップ1402にて作業定義「PD3:A3」及び「PD3:B3」を得る。ステップ1403において、これらの作業定義に属する実行要求の中で最も発生日時の古いものを1つ検索し、作業定義「PD3:A3」で案件名「P3」の実行要求（図11中の上から3番目のレコード）を取得する。次に、ステップ1405において、取得した実行要求に対する自動実行業務プログラム0103の業務処理実行を開始し、ステップ1406において実行カウンタを1加算する。これにより、スケジューリング制御単位「SC2」の実行カウンタは3となり、最大実行多重度3と比較して同時実行数に余裕がなくなった為（ステップ1407）、図14のフローチャートの処理を完了する。

【0052】

図13に戻って、ステップ1305においてスケジューリング制御単位「SC2」をチェック済みにマークし、ステップ1306にて未チェックのスケジューリング制御単位「SC3」の存在を確認する。再び、ステップ1302にて「SC3」を選択し、現在の実行数カウント4と最大実行多重度5から実行多重度に余裕がある為、ステップ1304にてスケジューリング制御単位「SC3」の実行要求に対する業務処理を開始する。再び、ステップ1304内の処理手順を図14を用いて示す。

【0053】

先ず、ステップ1401にて優先度レベル「高」を選択し、ステップ1402にてスケジューリング制御単位「SC3」に属する優先度レベル「高」の作業定義リストを検索する。ここでは、検索結果として作業定義「PD2:C2」を得る。次に、ステップ1403において、この作業定義に属する実行要求の中で最も発生日時の古いものを1つ検索し、案件名「P5」の実行要求（図11中の上から5番目のレコード）を得る。次に、ステップ1405において、取得した実行要求に対する自動実行業務プログラム0103の業務処理実行を開始し、ステップ1406において実行カウンタを1加算する。これにより、スケジューリング制御単位「SC3」の実行カウンタは5となり、最大実行多重度と比較して同

時実行数に余裕が無くなった為（ステップ1407）、図14のフローチャートの処理を完了する。

【0054】

再び、図13に戻って、ステップ1305においてスケジューリング制御単位「SC3」をチェック済みにマークし、ステップ1306にて未チェックのスケジューリング制御単位の存在をチェックする。ここでは、既に全てのスケジューリング制御単位がチェック済みであり、図13のフローチャートに示す「実行計画処理」を完了する。

【0055】

上記に示したように、業務プログラム実行制御部0112は、業務プログラム実行スケジュール定義0712に従って、適切に自動実行業務プログラム0103による業務処理の実行を制御することが出来る。

【0056】

次に、プロセス定義及び業務プログラム実行スケジュール定義の別の定義例を用いて、本発明により具体的にどのように課題が解決されるかを示す。

【0057】

図15に示す定義例では、4つの作業定義「A」、「B」、「C」及び「D」を順に実行するプロセス定義「PD4」を用いる。ここで、作業定義「A」、「B」及び「D」は作業種別が「自動実行」であり、作業定義「C」は作業種別が「作業員実行」である。また、作業定義「A」は自動実行業務プログラム「Program1」により業務処理を行い、作業定義「B」及び「C」は共に同じ自動実行業務プログラム「Program2」により業務処理を行うものとする。また2つの自動実行業務プログラム「Program1」及び「Program2」は同じ計算機上で動作する。この時、自動実行業務プログラム0103による業務処理の実行スケジュールリングにおける要件として、以下の2点が挙げられる。第1に、全ての自動実行業務プログラム0103は同一計算機0131上で動作する為、これら全体での処理負荷を一定に抑える必要がある。第2に、作業員（人）による業務処理はスループットが低い為、新規に投入した案件が速やかに（作業員により業務処理が行われる）作業定義「C」へ進んで行く必要がある

。それにより、作業者が業務処理に着手する時刻を早めることが出来、業務フロー全体としての処理効率が向上する。例えば、バッチ処理等により一時に多数の案件が投入される場合、自動実行業務プログラム0103による業務処理を作業の発生順に実行すると、作業定義「A」における一連の実行要求に対する処理が終わらない限り業務フローが先に進まない事になり、業務フロー全体で見ると非効率となる。これらの要件を満たす為には、以下の様に業務プログラム実行スケジュール定義を定めるだけでよい。先ず、全ての「自動実行」の作業を1つのスケジューリング制御単位「SC4」として定義する。これにより、第1の要件が満たせる。次に第2の要件を満たす為、案件を速やかに作業定義「C」へ流す目的で、作業定義「A」から作業定義「B」にかけて下流に行く程優先度が高くなるように設定する。これにより、作業定義「C」に向けてより下流に進んだ案件から、優先的に自動実行業務プログラム0103による業務処理が行われる為、速やかに作業定義「C」へ案件が流れることになる。本実施例では、優先度のレベルとして「低」、「中」、「高」の3段階しか指定できないが、実装によっては優先度のレベル分けをより細かく分ける事も可能である。また、作業定義「D」における業務処理は、プロセス定義の最下流であって、人である作業者による業務処理の実行には影響を与えない。その為、例えば一日の業務の最後で纏めて処理を行えばよい。そのため、作業定義「D」の優先度を下げる事で「作業者実行」の作業定義「C」より上流の業務処理に計算機リソースを優先的に使うことが出来る。以上により、第2の要件を満たすことが出来る。こうした優先度設定は、図8に示したスケジューリング制御単位「SC3」の様に、異なるプロセス定義間でも柔軟に行うことが出来る。

【0058】

図16に示す定義例では、図15のプロセス定義「PD4」に加えて、「PD4」の一部を変更したプロセス定義「PD5」を追加している。プロセス定義「PD5」は、作業種別が「作業者実行」の作業定義「C'」を追加した以外は、プロセス定義「PD4」と同じである。その為、作業定義「PD4：A」及び「PD5：A」は共に自動実行業務プログラム「Program1」により業務処理を行い、作業定義「PD4：B」、「PD4：D」、「PD5：B」、及び「PD5：D」は

全て自動実行業務プログラム「Program2」により業務処理を行う。また、自動実行業務プログラム「Program1」及び「Program2」は同一計算機上で動作している。またこの時、図15の例と同様に、プロセス定義「PD4」上には作業定義「PD4:A」、「PD4:B」、及び「PD4:D」からなるスケジューリング制御単位「SC4'」を定義し、プロセス定義「PD5」にも同様のスケジューリング制御単位「SC5」を定義する。更に、図15の例においてスケジューリング制御単位「SC4」に割当てた最大実行多重度5を、図16の例では2つのスケジューリング制御単位「SC4'」及び「SC5」の間で分配し、それぞれ最大実行多重度3（「SC4'」）と2（「SC5」）とする事で、自動実行業務プログラム0103による業務処理を行う計算機リソースを2つのプロセス定義間で固定的に分配し、負荷制御する事が可能である。

【0059】

以上2つの定義例に見てきたように、本発明によれば、自動実行業務プログラムによる業務処理の実行スケジューリングや負荷制御に関する設定を、業務フローのモデルであるプロセス定義に基づいて直感的かつ容易に行う事ができ、また、業務フローモデルに固有の実行スケジューリングが可能となる。

【0060】

また、本発明によれば、業務プログラム単位ではなく作業定義単位で、自動実行業務プログラムによる業務処理の実行スケジューリング方法の指定が可能である。これにより、異なる作業定義間で自動実行業務プログラムを共用している場合でも、各作業定義単位で柔軟且つ詳細に実行スケジューリングや負荷制御の設定を行う事が出来る。また、異なる業務フローを跨って、スケジューリング設定を行うことも可能である。

【0061】

例えば、図15の定義例に見るように、業務フローモデルに関連したスケジューリングの要件に応じて、同一の自動実行業務プログラムを共用する複数の作業定義間で処理の優先度を割付ける事が可能である。

【0062】

また、図16の定義例に見るように、異なる業務フロー間で自動実行業務プロ

グラムを共用したり、もしくは業務プログラムの処理を行う計算機を共用したりする場合でも、業務フロー毎に個別に最大実行多重度を指定する事が可能であり、業務フロー間での計算機リソースの競合による相互干渉を回避する事が出来る。

【 0 0 6 3 】

こうした、ワークフロー管理システムにおける業務プログラムの実行スケジューリングや負荷制御は、特に近年の以下の様な技術的背景において重要度が増していると言える。

【 0 0 6 4 】

例えば、E A I (Enterprise Application Integration) 分野においてワークフロー管理システムの技術を業務システム間連携に用いたり (EAI Journal, September/October 1999, pp.49-53)、より複雑な業務手順を有する基幹系業務への適用が進むにつれ、ワークフロー管理システムにおいて人手を介さず業務プログラムの実行制御を行う場合が増えてきている。それに伴い、様々な業務フローの混在する高トラフィックなワークフロー実行環境において、業務プログラム処理の安定した実行制御を行う必要があり、また実行スケジューリングや負荷制御の設定定義を適切かつ容易に出来ることが必要となって来ている。

【 0 0 6 5 】

また近年、A S P (Application Service Provider) 等のアプリケーションホスティングサービスが注目されて来っており (日経インターネットテクノロジー、1999年8月号、pp.98-105)、ワークフロー管理システムを用いたアプリケーションにおいても同様のサービス提供形態が考えられる。アプリケーションホスティングサービスとして提供する場合、顧客や業務毎に使用する計算機リソース量の管理を適切に行い、制限したり課金したりする必要があるが、本発明はこうした適用分野においても有効である。

【 0 0 6 6 】

以下に、A S P サービスへの本発明の適用の一実施例を示す。

【 0 0 6 7 】

本実施例では、次の2通りの方法でサービス品質を定義し、品質に応じた課金

をユーザに対して行えるようにする。即ち、第1のサービス提供方法として、ユーザの各契約において使用可能な最大計算機リソース量を固定的に定め、常に設定した計算機リソース量の範囲でのサービス提供を保証する。この場合、設定した計算機リソース量に応じて、提供サービスに対する課金を行う。第2のサービス提供方法は、固定の計算機リソース量を複数ユーザ間で共有するもので、各契約に対し設定した処理優先度に応じて動的に共有する計算機リソースの配分を行う方法である。この場合、割当ての処理優先度に応じてサービス料金を課金する。この時、ユーザは常に一定サービス量の提供が保証される訳ではないが、例えば業務処理の緊急度の低いユーザは優先度を下げて契約する事で、コストを抑える事が出来る。但しこの時、ASPによるサービス提供は、夜間等の計算機リソースに余裕のある時間帯に回される事になる可能性がある。

【0068】

本実施例では、ユーザの各契約に対しプロセス定義を用意し、契約とプロセス定義とを対応付けて管理する。異なるユーザに対して提供する業務サービスが同様な業務フローであっても、ユーザ毎に個別のプロセス定義を定義する必要があるが、後々ユーザ毎のカスタマイズを行う必要性があることを考慮すると、プロセス定義を契約毎に分けておくのは自然である。また簡単の為、本実施例では業務フロー全体で使用する計算機リソース量を課金対象とし、業務フローの部分的な作業定義に対して使用計算機リソース量を制御する事はしないものとする。

【0069】

図17に、本実施例におけるシステム構成を示す。このシステム構成では、図1及び図7に示したシステム構成において、更にユーザ管理情報1711及び運用定義ツール1701を追加している。図17中には差分のみ記載しており、システム構成全体の詳細に関しては記載していない。ユーザ管理情報1711は、ASPの提供するサービスに対する各ユーザの契約内容を保持しており、ユーザの各契約に対するサービスの提供品質及び課金の為の管理情報として用いられる。また、このテーブルに格納されている契約内容に合わせて業務プログラム実行スケジューリング定義0712も更新される。これらの情報は、運用定義ツール1701により編集される。運用定義ツール1701は、ソフトウェアプログラ

ムであり、他のソフトウェアプログラムと同様に計算機0131上で動作する。

【0070】

図18は、運用定義ツール1701の表示画面例である。契約リスト1801の各行は、業務サービスの各契約を表している。各契約において、ユーザ名1811は該契約の契約ユーザ名を、プロセス定義名1812は該契約で使用するプロセス定義を、契約形態1813は上述したサービス提供方法の違いにより「固定」か「共有」の何れかの契約形態を表し、優先度1804は契約形態1813が「共有」の場合の共有グループ内での処理優先度を表す。また、実行多重度1815は、契約形態1813が「固定」の場合その契約に割当てられる処理実行多重度を、契約形態1813が「共有」の場合、共有グループ全体に割当てられる処理実行多重度を表す。その為、契約形態1813が「共有」の全契約に対して実行多重度1815の値は同じで、何れかの契約において値を変更した場合は、他の全ての「共有」契約形態の契約に対して変更後の値が反映される。図中では、共有グループに対する実行多重度1815は「10」となっている。追加ボタン1821、削除ボタン1822により、契約リスト1801へ新たな契約を追加したり、既存の契約を選択して削除する事が可能である。追加時には、編集画面1802が表示され、各設定項目を入力後、OKボタン1831により設定の追加を確定する。また終了ボタン1823により、運用定義ツール1701を終了させることが出来る。契約リスト1801への新規契約追加や削除時には、運用定義ツール1701はユーザ管理情報1711、及び業務プログラム実行スケジュール定義0712の更新処理を行う。この更新処理の詳細に関しては、後述する。

【0071】

図19に、ユーザ管理情報1711の補助記憶装置0154への格納形式を示す。ユーザ管理情報テーブル1901の各カラム、ユーザ名1911、プロセス定義名1912、契約形態1913、優先度1914、実行多重度1915は、夫々運用定義ツール1701の表示画面における契約リスト1801の各カラムに対応し、契約リスト1801に表示される各契約の内容がそのままユーザ管理情報テーブル1901の各行として格納される。

【0072】

図20は、ユーザ管理情報1711の内容が図19に示す各レコードの通りの場合に、それに対応して業務プログラム実行スケジュール定義0712内に格納される設定情報を示す。図20中のスケジューリング制御単位定義情報テーブル0911には、「共有」契約形態の全契約において共有される単一のスケジューリング制御単位「SC-S」と、「固定」契約形態の各契約夫々に対して設定される複数のスケジューリング制御単位「SC-P1」「SC-P2」「SC-P3」とを含んでいる。スケジューリング制御単位「SC-S」は、「共有」契約形態の全ての契約に対するプロセス定義から共有されるスケジューリング制御単位である。スケジューリング制御単位作業定義情報テーブル1011において、スケジューリング制御単位名1001が「SC-S」である各行は、これらの「共有」契約形態のプロセス定義であり、プロセス定義「APD4」「APD5」「APD6」がこれに当たる。また、作業定義名1003において、「*」はプロセス定義内の「自動実行」種別の全作業が指定されたスケジューリング制御単位に含まれる事を意味する。すなわち、プロセス定義「APD4」「APD5」「APD6」内の全ての「自動実行」種別の作業は、スケジューリング制御単位「SC-S」に含まれ、最大実行多重度「10」に制限された計算機リソースにて実行される。次に、スケジューリング制御単位定義情報テーブル0911におけるスケジューリング制御単位「SC-P1」「SC-P2」「SC-P3」は、「固定」契約形態の契約に対するスケジューリング制御単位であり、各契約は夫々プロセス定義「APD1」「APD2」「APD3」に対応する。例えば、スケジューリング制御単位「SC-P1」は、ユーザ「USR1」の契約するプロセス定義「APD1」の全ての「自動実行」種別の作業を含み、最大実行多重度「3」を提供する。

【0073】

以下、運用定義ツール1701によるユーザ管理情報1711及び業務プログラム実行スケジュール定義1812の編集処理手順の詳細について述べる。最初に、新規契約追加時の処理手順について述べる。まず、運用定義ツール1701で指定した各設定内容に従って、ユーザ管理情報テーブル1901へレコードを追加する。次に、スケジューリング制御単位作業定義情報テーブル1011へ、

運用定義ツール 1 7 0 1 で指定されたプロセス定義のレコードを挿入する。この時、作業定義名 1 0 0 3 へは、同プロセス定義内の全「自動実行」種別の作業を示す「*」を設定し、プロセス定義名 1 0 0 2、優先度 1 0 0 4 へは運用定義ツール 1 7 0 1 にて指定された値を設定する。また、スケジューリング制御単位名 1 0 0 1 へは、追加する新規契約の契約形態が「共有」の場合はスケジューリング制御単位「SC-S」を、「固定」の場合はスケジューリング制御単位定義情報テーブル 0 9 1 1 へ新規にスケジューリング制御単位を追加し、その名称を設定する。この時、新規に追加したスケジューリング制御単位の最大実行多重度 0 9 0 2 には、運用定義ツール 1 7 0 1 にて指定した実行多重度 1 8 1 5 を設定する。更に、契約形態が「共有」の場合、実行多重度 1 8 1 5 の値をスケジューリング制御単位「SC-S」の最大実行多重度 0 9 0 2 に設定し、ユーザ管理情報テーブル 1 9 0 1 内の契約形態が「共有」の全てのレコードの実行多重度 1 9 1 5 カラムにも設定する。次に、契約削除時の処理手順について述べる。まず、運用定義ツール 1 7 0 1 にて指定した契約に対するレコードを、プロセス定義名 1 9 1 2 をキーに検索し、ユーザ管理情報テーブル 1 9 0 1 から削除する。同様に、プロセス定義名 1 0 0 2 をキーに、スケジューリング制御単位作業定義情報テーブル 1 0 1 1 から対応するレコードを削除する。この時、削除する契約の契約形態が「固定」の場合、併せてスケジューリング制御単位定義情報テーブル 0 9 1 1 から該プロセス定義に対応するスケジューリング制御単位を削除する。

【 0 0 7 4 】

この様にして設定定義した業務プログラム実行スケジュール定義 0 7 1 2 に従って、どの様に各契約に対して提供するサービス品質を制御するかは、第一の実施例において既に示した通りである。以上に示したように、本発明を A S P サービスに適用することにより、ユーザの各契約に対しサービス品質を制御し、サービス品質に応じて課金を行う事が可能となる。

【 0 0 7 5 】

次に、単純な実行多重度による制御のみでなく、より一般化した計算機リソース使用量による業務プログラム実行スケジューリングの実施例を示す。計算機リソースの例としては、メモリ使用量、ディスク使用量、CPU使用量、データベ

ースアクセス量／コネクション数、ネットワーク使用量、等がある。ここでは、複数種類の計算機リソースの使用制限による業務プログラム実行スケジューリング制御の一例として、メモリ使用量（１種類目の計算機リソース）とディスク使用量（２種類目の計算機リソース）を組み合わせる例を示す。説明の冗長を避ける為、ここでは第一の実施例との差分のみ示すこととする。

【 0 0 7 6 】

図 2 1 は、各業務プログラムの使用する計算機リソース量を予め設定し格納した、業務プログラム使用リソース情報テーブル 2 1 0 1 を示す。業務プログラム使用リソース情報テーブル 2 1 0 1 は、補助記憶装置 0 1 5 4 に格納され、業務プログラム実行計画部 0 7 0 2 により参照される。業務プログラム使用リソース情報テーブル 2 1 0 1 の各カラムは、業務プログラムを特定する為の業務プログラム情報 2 1 1 1、該業務プログラムの使用するメモリ使用量 2 1 1 2、及びディスク使用量 2 1 1 2 とから成る。図 2 1 中では、例えば業務プログラム「Program 1」は一つの業務処理の実行時にメモリ「1MB」及びディスク「10MB」を使用する。他の業務プログラムに関しても同様に、「Program 2」はメモリ「10MB」ディスク「100MB」を、「Program 3」はメモリ「50MB」ディスク「50MB」を、「Program 4」はメモリ「30MB」ディスク「200MB」を使用する。

【 0 0 7 7 】

図 2 2 は、各スケジューリング制御単位において使用可能な最大使用リソース量を管理する拡張スケジューリング制御単位定義情報テーブル 2 2 0 1 である。本実施例では、最大メモリ使用量 2 2 1 2、及び最大ディスク使用量 2 2 1 3 を、各スケジューリング制御単位に定めており、スケジューリング制御単位名 2 2 1 1 により特定できる。拡張スケジューリング制御単位定義情報テーブル 2 2 0 1 は、第一の実施例におけるスケジューリング制御単位定義情報テーブル 0 9 1 1 に置換えて用いる。図 2 2 の例では、スケジューリング制御単位「SC 1」に対しては最大メモリ使用量「1GB」最大ディスク使用量「4GB」を、スケジューリング制御単位「SC 2」に対しては最大メモリ使用量「500MB」最大ディスク使用量「2GB」を、スケジューリング制御単位「SC 3」に対しては

最大メモリ使用量「200MB」最大ディスク使用量「1GB」を、割当てている。各スケジューリング制御単位に属する作業において自動実行業務プログラム0103による業務処理実行は、スケジューリング制御単位の全体で上記設定された最大メモリ使用量、最大ディスク使用量の範囲を超えないように、業務プログラム実行計画部0702により制御される。

【0078】

本実施例では、業務プログラム実行計画部0702の機能を拡張し、実行多重度のみではなく、複数の一般化した計算機リソース量の範囲内で業務プログラムの実行計画を行うように変更する。実行多重度も、このような一般化した計算機リソースの1つとして考えられる為、この拡張のなかで他の計算機リソースと同様に扱える。以下に、複数の一般化された計算機リソースに対して実行計画を行うように機能拡張した業務プログラム実行計画部0702の処理手順を詳細に説明する。

【0079】

まず、業務プログラム実行計画部0702の「初期処理」の処理手順について説明する。第一の実施例で述べたように、業務プログラム実行計画部0702の「初期処理」は、ワークフロー実行制御エンジン0101の起動時に一度のみ実行される。本処理ではまず、業務プログラム実行スケジュール定義0712内の拡張スケジューリング制御単位定義情報テーブル2201を検索し、スケジューリング制御単位のリストを取得する。次に、各スケジューリング制御単位に対し、各計算機リソースの現在の使用リソース量を保持する変数領域を確保し、現在の使用リソース量を0に初期化する。ここでは、メモリ使用量、及びディスク使用量の値を保持する2つの変数領域を用意する。最後に、業務プログラム実行計画部0702の実行計画処理起動トリガを起す。

【0080】

第一の実施例で述べたように、業務プログラム実行計画部0702は実行計画処理起動トリガの発生を受けて、「実行計画処理」を開始する。図23～図24のフローチャートを用いて、「実行計画処理」の詳細な処理手順を示す。

【0081】

図23に示す処理手順では、スケジューリング制御単位のリストを取得し、各スケジューリング制御単位において使用可能な最大計算機リソース量の範囲での自動実行業務プログラムの実行要求を開始する。以下、各ステップを追って詳細に説明する。（ステップ2301）業務プログラム実行スケジュール定義0712内の拡張スケジューリング制御単位定義情報テーブル2201を検索し、スケジューリング制御単位のリストを取得する。このとき、各計算機リソースの最大使用量（最大メモリ使用量、及び最大ディスク使用量）も併せて取得する。（ステップ2302）ステップ2301で取得したスケジューリング制御単位のリストから、未チェックのスケジューリング制御単位を1つ選択する。（ステップ2303）当該スケジューリング制御単位における自動実行業務プログラムの実行要求を、使用可能な最大計算機リソース使用量の範囲内で実行開始する。このステップの処理は、図24のフローチャートにおいて後ほど詳しく説明する。（ステップ2304）ステップ2302にて選択したスケジューリング制御単位を、チェック済みにマークする。（ステップ2305）未チェックのスケジューリング制御単位が残っている場合、ステップ2302へ戻り、同様の処理を繰り返す。全てのスケジューリング制御単位に対する処理が完了したら、「実行計画処理」を終了する。

【0082】

次に、図24を用いて、図23のフローチャートにおけるステップ2303の処理を詳細に説明する。即ち図24は、ステップ2302で選択した単一スケジューリング制御単位における、業務処理の実行要求に対する実行計画処理を詳述するものである。以下、各ステップを追って詳細に説明する。（ステップ2401）未処理の優先度レベルを、優先度の高い順に1つ選択する。（ステップ2402）図23のステップ2303において選択されているスケジューリング制御単位に属し、且つステップ2401で選択した優先度レベルの作業定義のリストを、スケジューリング制御単位作業定義情報テーブル1011より検索する。（ステップ2403）ステップ2402で検索した作業定義のリストを指定して、業務プログラム実行要求管理部0701の「実行要求取得処理」を呼出す。これ

により、指定した作業定義に属する実行要求の内、発生日時の最も古いものを1つ取得できる。(ステップ2405) ステップ2403にて実行要求が取得できた場合、取得した実行要求に対する業務プログラムの使用計算機リソース量2112及び2113を業務プログラム使用リソース情報テーブル2101から取得し、残計算機リソース量と比較する。即ち、変数領域に保持している現在の使用計算機リソース量と、取得した実行要求に対する業務プログラムの使用計算機リソース量とを加算し、当該スケジューリング制御単位において使用可能な最大計算機リソース量を超過するか否かを判定する。これを、計算機リソースの種類毎(本実施例では、メモリ使用量、及びディスク使用量の2種類)に対して行い、1種類でも超過するものがあるか否かを調べる。(ステップ2407) ステップ2405での比較の結果、全種類の計算機リソースにおいて、最大計算機リソース量を超過しない場合、ステップ2403にて取得した実行要求に対する自動実行業務プログラム0103による業務処理の実行を、業務プログラム起動制御部0703へ依頼する。(ステップ2408) 当該スケジューリング制御単位における現在の使用計算機リソース量を最新の状態に更新する。即ち、ステップ2405での比較に用いた、加算後の計算機リソース量を変数領域に保存する。その後、ステップ2403へ戻り処理を繰返す。(ステップ2411) ステップ2405での比較の結果、1種類でも超過するものが存在する場合、業務プログラム実行要求管理部0701を呼出してステップ2403で取得した処理要求を業務プログラム実行要求管理テーブル1111へ戻す。その後、ステップ2303の処理を完了する。(ステップ2409) ステップ2403にて、選択した優先度に対する実行要求が存在しなかった場合、当該優先度を処理済にマークする。(ステップ2410) 未処理の優先度が存在する場合、ステップ2401へ戻り、次に高い優先度レベルを選択し直して処理を繰返す。

【0083】

更に、業務プログラム実行要求管理部0701においては、ステップ2411で見たように、一旦取得した処理要求を業務プログラム実行要求管理テーブル1111へ戻す処理を拡張する必要がある。これは、単純に業務プログラム実行計画部0702から受取った処理要求を業務プログラム実行要求管理テーブル11

1 1 へ挿入するだけである。

【 0 0 8 4 】

以上に示したように、本実施例では第一の実施例を拡張することで、実行多重度のみでなく一般化した計算機リソースの使用量を制限する事で、自動実行業務プログラムによる業務処理の実行計画を行う事が出来る。この一般化においては、第一の実施例における実行多重度による制御も、1 計算機リソースの種類として扱える。

【 0 0 8 5 】

【発明の効果】

以上に述べたように、作業に対する処理を実行する業務プログラムが使用する計算機リソースの配分を考慮したワークフロー管理が実現できる。

【図面の簡単な説明】

【図 1】

本発明の一実施例におけるワークフロー管理システムのシステム構成図である。

【図 2】

本発明の一実施例におけるワークフロー管理システムの、プロセス定義の例である。

【図 3】

図 1 に示すプロセス定義 0 1 2 1 の、定義情報格納テーブルの構成を示す図である。

【図 4】

図 1 に示すプロセス定義 0 1 2 1 の、定義情報格納テーブルの構成を示す図である。

【図 5】

図 1 に示すプロセス定義 0 1 2 1 の、定義情報格納テーブルの構成を示す図である。

【図 6】

図 1 に示すワークフロー実行状態情報 0 1 2 2 の、格納テーブルの構成を示す

図である。

【図 7】

図 1 における業務プログラム実行制御部 0 1 1 2 の、詳細な構成を示す図である。

【図 8】

業務プログラム実行スケジュール定義 0 7 1 2 の定義例を示す図である。

【図 9】

業務プログラム実行スケジュール定義 0 7 1 2 の、定義情報格納テーブルの構成を示す図である。

【図 1 0】

業務プログラム実行スケジュール定義 0 7 1 2 の、定義情報格納テーブルの構成を示す図である。

【図 1 1】

業務プログラム実行要求 0 7 1 1 の、格納テーブルの構成を示す図である。

【図 1 2】

業務プログラム実行要求管理部 0 7 0 1 の、実行要求取得手続きの動作を示すフロー図である。

【図 1 3】

ワークフロー実行計画部 0 7 0 2 の、実行計画処理の動作を示すフロー図である。

【図 1 4】

図 1 3 のステップ 1 3 0 4 の詳細を示すフロー図である。

【図 1 5】

業務プログラム実行スケジュール定義 0 7 1 2 の定義例を示す図である。

【図 1 6】

業務プログラム実行スケジュール定義 0 7 1 2 の定義例を示す図である。

【図 1 7】

A S P サービスへの本発明の適用を示した一実施例における、システム構成図である。

【図 1 8】

運用定義ツール 1 7 0 1 の表示画面を示す図である。

【図 1 9】

図 1 7 に示すユーザ管理情報 1 7 1 1 の、格納テーブルの構成を示す図である。

【図 2 0】

A S P サービスへの本発明を適用を示した一実施例における、業務プログラム実行スケジュール定義 0 7 1 2 の定義例を示す図である。

【図 2 1】

一般的な計算機リソースを制限する実施例における、業務プログラム使用リソース情報テーブル 2 1 0 1 の構成を示す図である。

【図 2 2】

一般的な計算機リソースを制限する実施例における、拡張スケジューリング制御単位定義情報テーブル 2 2 0 1 の構成を示す図である。

【図 2 3】

一般的な計算機リソースを制限する実施例における、ワークフロー実行計画部 0 7 0 2 の、実行計画処理の動作を示すフロー図である。

【図 2 4】

図 2 3 のステップ 2 3 0 2 の詳細を示すフロー図である。

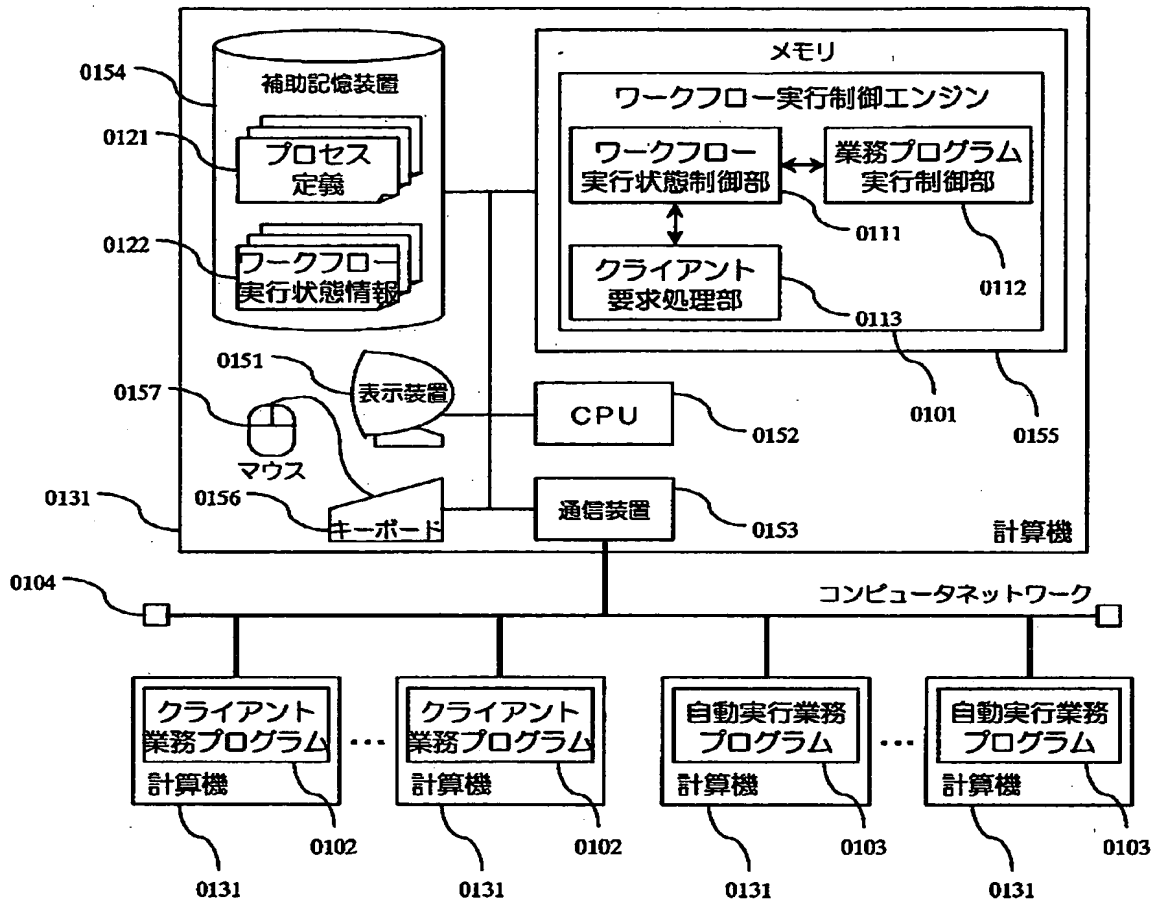
【符号の説明】

0 1 0 1 …ワークフロー実行制御エンジン、0 1 0 2 …クライアント業務プログラム、0 1 0 3 …自動実行業務プログラム、0 1 0 4 …コンピュータネットワーク、0 1 1 1 …ワークフロー実行状態制御部、0 1 1 2 …業務プログラム実行制御部、0 1 1 3 …クライアント要求処理部、0 1 2 1 …プロセス定義、0 1 2 2 …ワークフロー実行状態情報、0 1 3 1 …計算機、0 7 0 1 …業務プログラム実行要求管理部、0 7 0 2 …業務プログラム実行計画部、0 7 0 3 …業務プログラム起動制御部、0 7 1 1 …業務プログラム実行要求、0 7 1 2 …業務プログラム実行スケジュール定義、0 8 0 1 …スケジューリング制御単位

【書類名】 図面

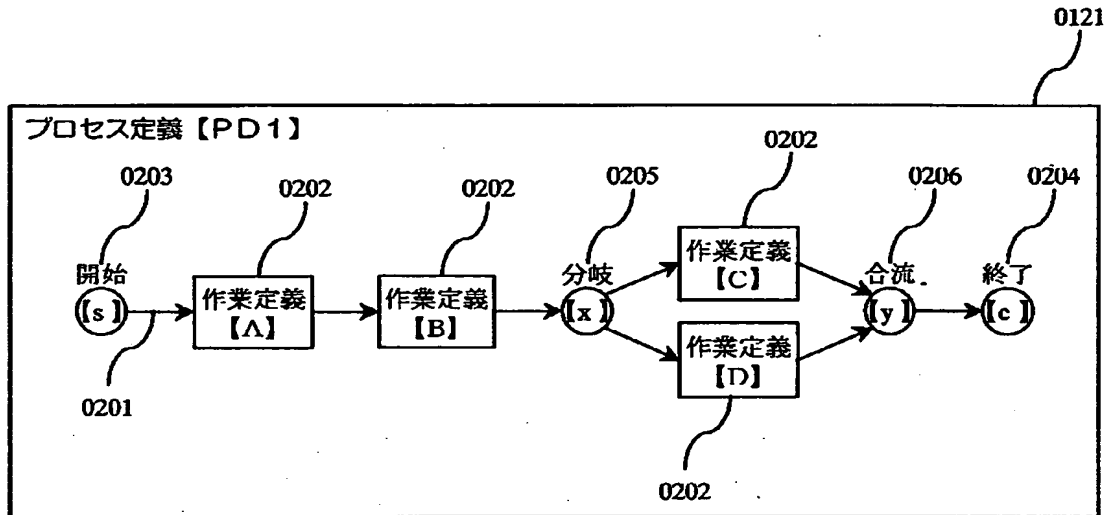
【図 1】

図 1



【図 2】

図 2



【図 3】

図 3

0301 プロセス定義名	0302 作業定義名	0303 作業種別	0304 作業者	0305 業務プログラム情報
PD1	A	作業者実行	山田	-
PD1	B	自動実行	-	Program1
PD1	C	自動実行	-	Program2
PD1	D	作業者実行	鈴木	-
PD2

0311

【図4】

図4

プロセス定義名	制御ノード名	ノード種別
PD1	s	開始
PD1	e	終了
PD1	x	分岐
PD1	y	合流
PD2

【図5】

図5

プロセス定義名	遷移元	遷移先
PD1	s	A
PD1	A	B
PD1	B	x
PD1	x	C
PD1	x	D
PD1	C	y
PD1	D	y
PD1	y	e
PD2

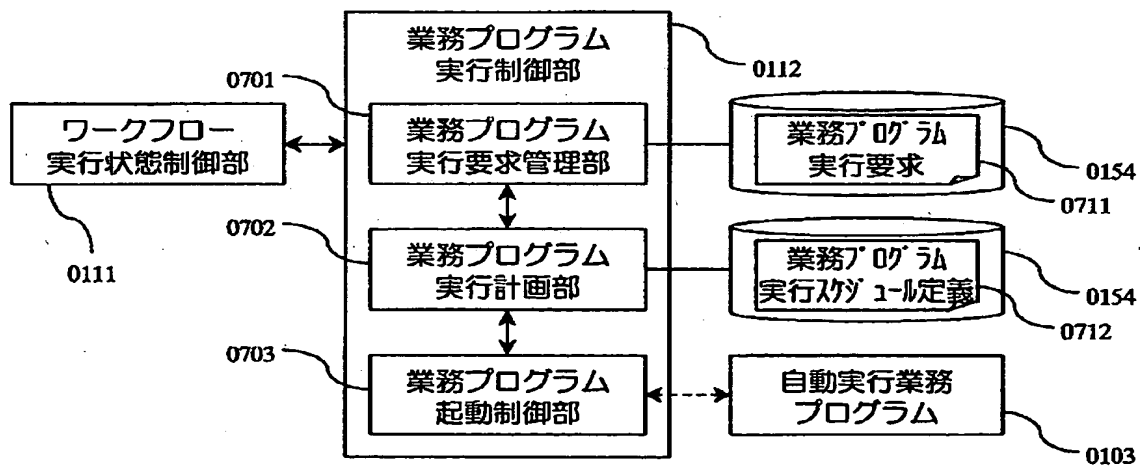
【図 6】

図 6

案件名	プロセス定義名	作業定義名	状態	作業者
P11	PD1	A	Completed	山田
P11	PD1	B	Ready	—
P11	PD1	C	Initial	—
P11	PD1	D	Initial	鈴木
P12	PD1	A	Initial	山田

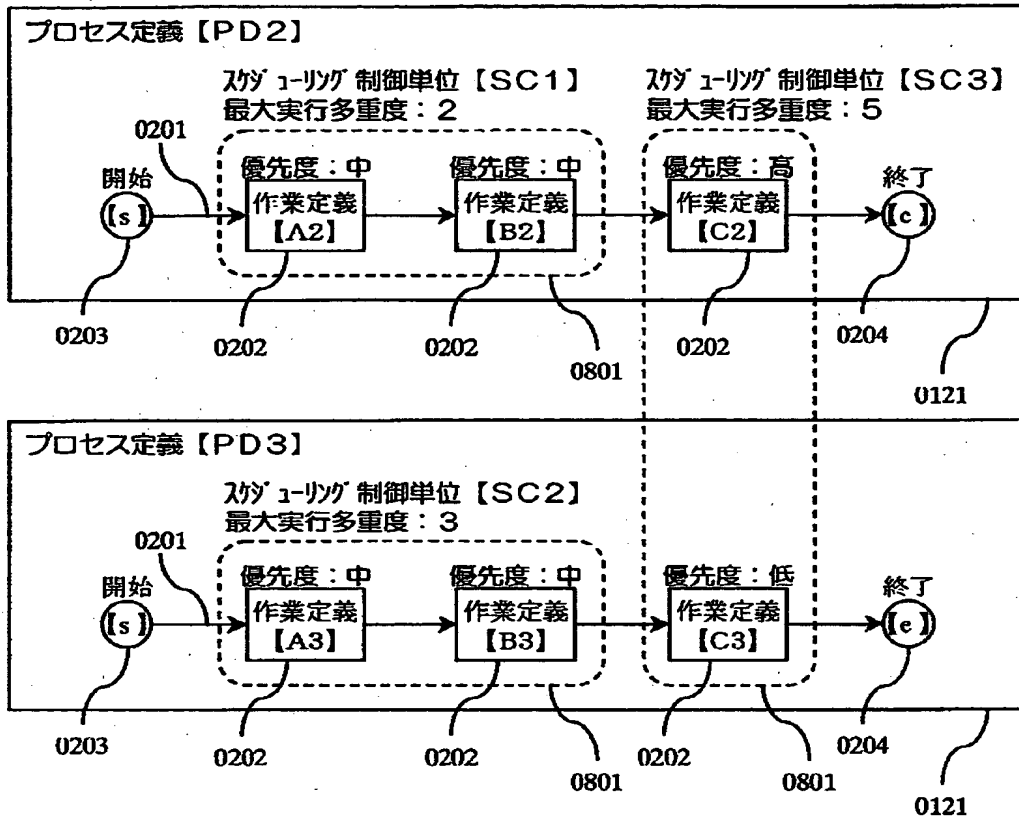
【図 7】

図 7



【図 8】

図 8



【図 9】

図 9

スケジューリング制御単位名	最大実行多重度
SC1	2
SC2	3
SC3	5

【図 1 0】

図 1 0

スケジューリング制御単位名	プロセス定義名	作業定義名	優先度
SC1	PD2	A2	中
SC1	PD2	B2	中
SC2	PD3	A3	中
SC2	PD3	B3	中
SC3	PD2	C2	高
SC3	PD3	C3	低

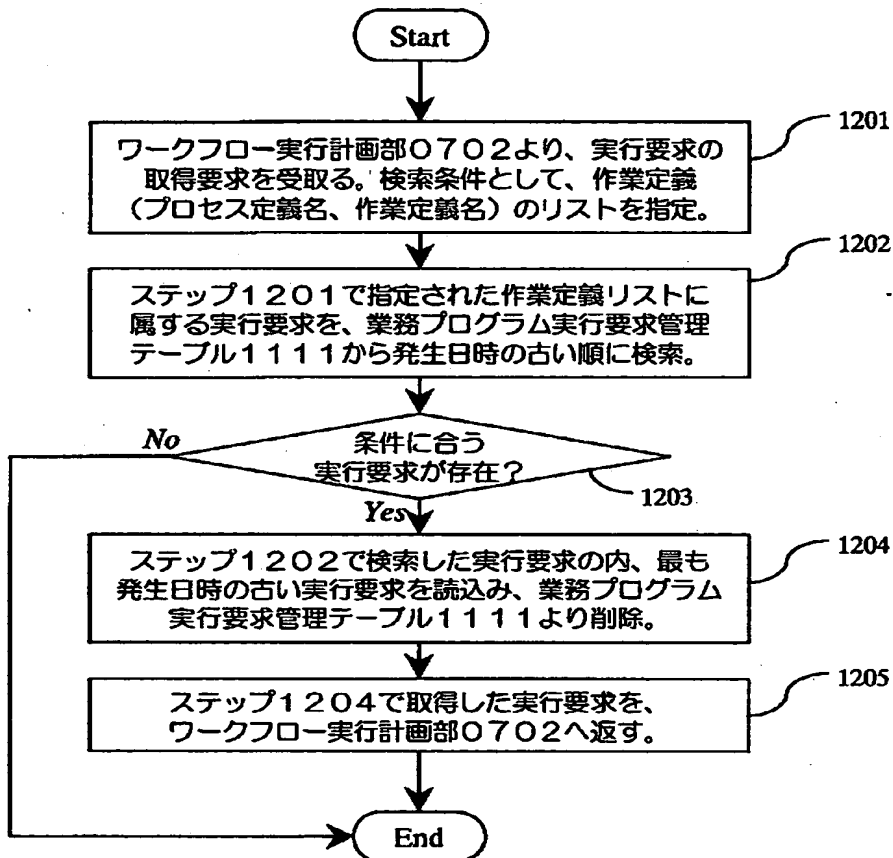
【図 11】

図 11

プロセス定義名	作業定義名	案件名	業務プログラム情報	発生日時
PD2	A2	P1	ProgramA	'00/08/08-12:00
PD2	B2	P2	ProgramB	'00/08/08-10:00
PD3	A3	P3	ProgramA	'00/08/07-12:00
PD3	B3	P4	ProgramB	'00/08/08-09:00
PD2	C2	P5	ProgramC	'00/08/09-12:00
PD3	C3	P6	ProgramC	'00/08/08-18:00

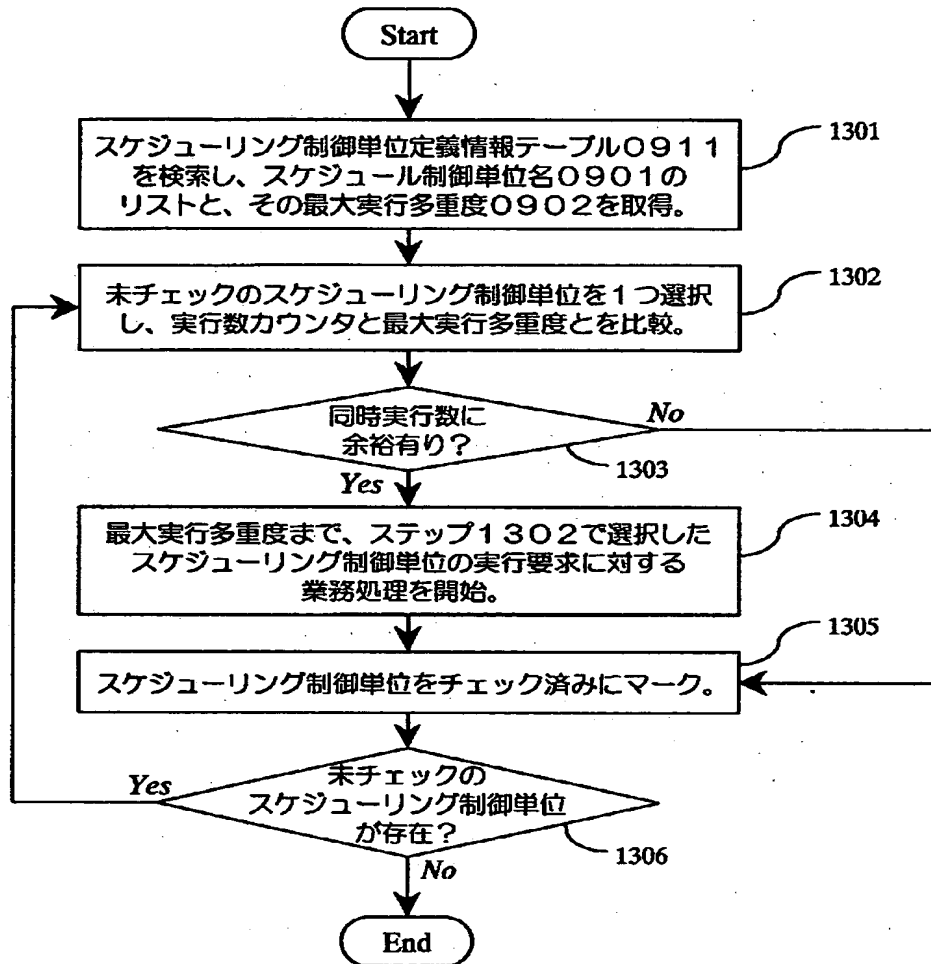
【図 1 2】

図 1 2



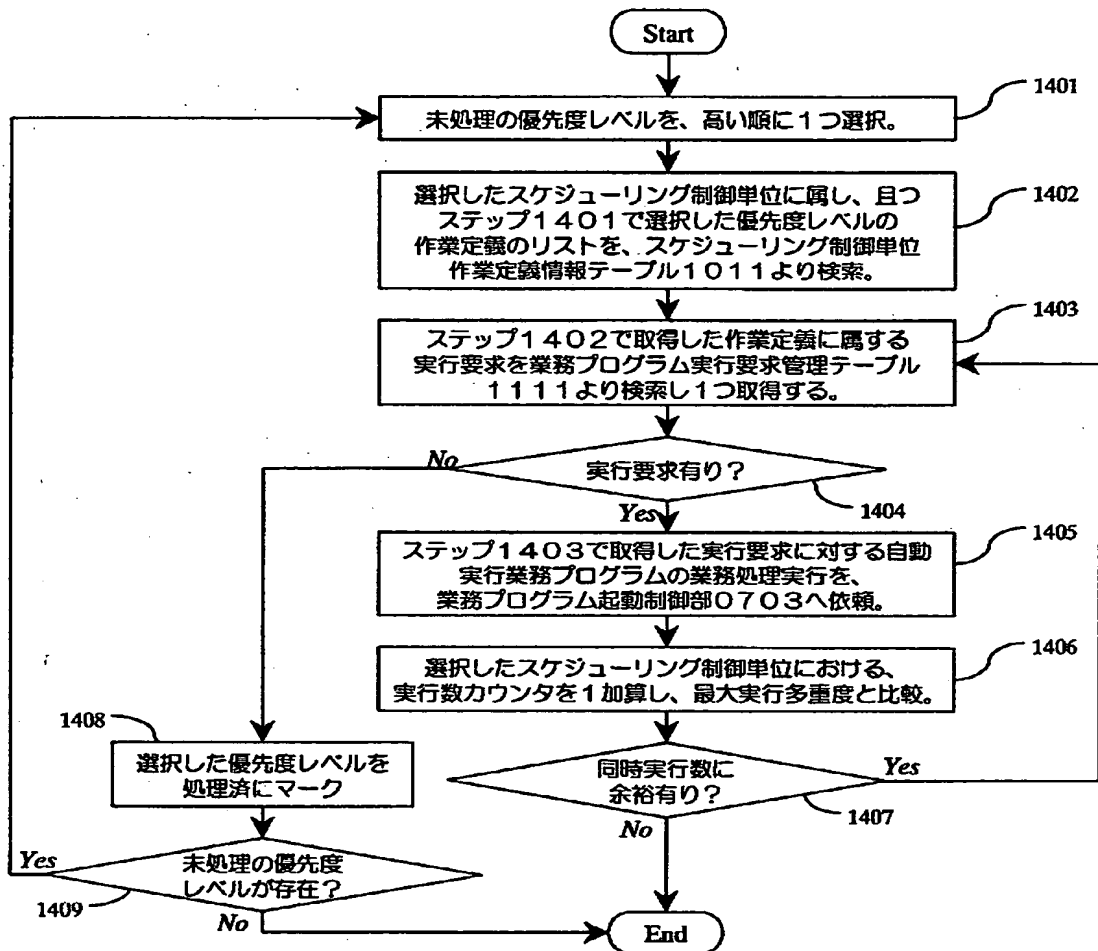
【図 1 3】

図 1 3



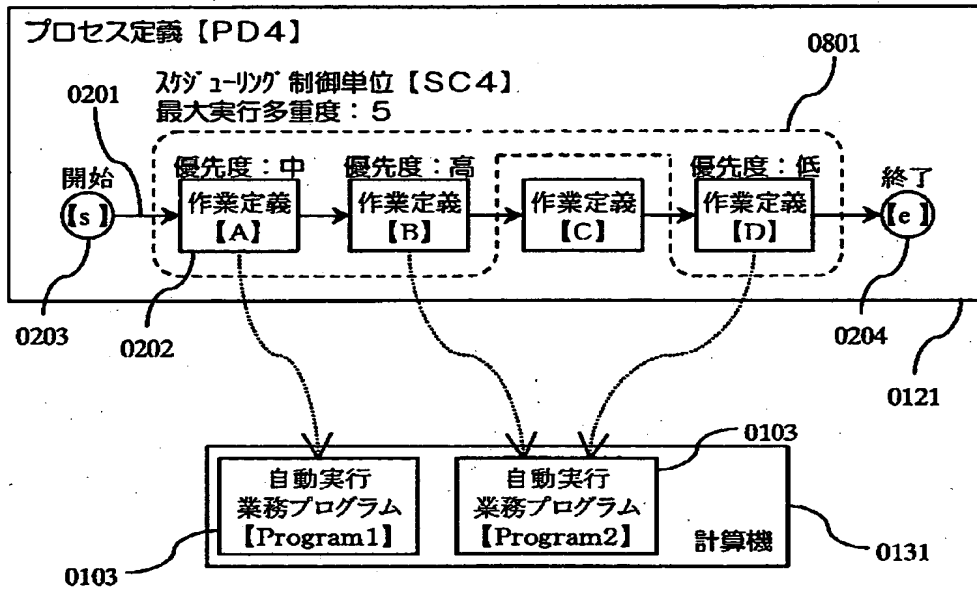
【図 14】

図 14



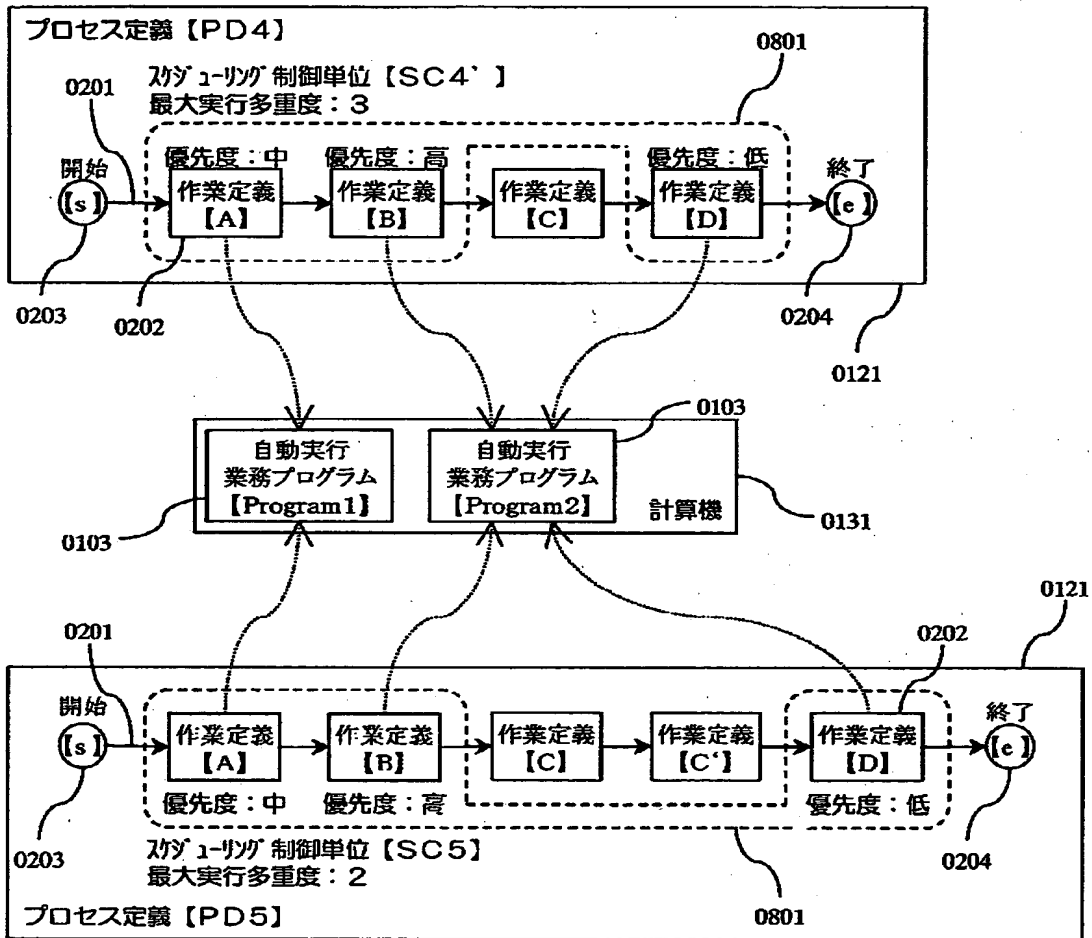
【図 1 5】

図 1 5



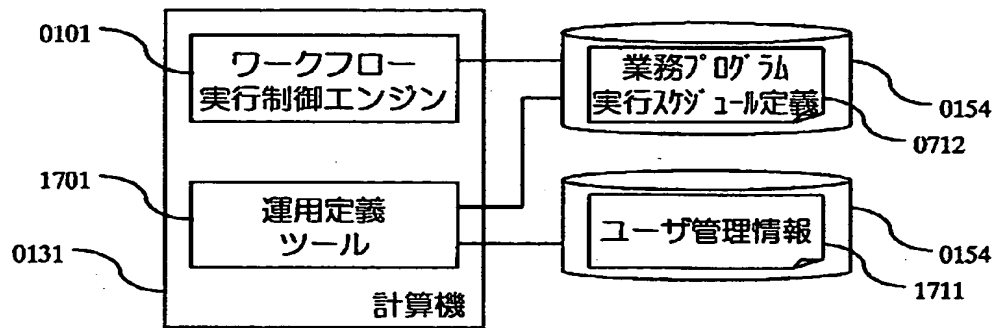
【図16】

図16



【図 17】

図 17



【図 1 8】

図 1 8

Figure 18 illustrates a user interface for managing process definitions. The main window (1801) contains a table (1811) with the following data:

ユーザ名	プロセス定義名	契約形態	優先度	実行多重度
USR1	APD1	固定	中	3
USR1	APD2	固定	中	5
USR2	APD3	固定	中	2
USR3	APD4	共有	中	10
USR4	APD5	共有	高	10
USR5	APD6	共有	低	10

Below the table are three buttons: 追加 (Add, 1821), 削除 (Delete, 1822), and 終了 (End, 1823). To the right of the main window is a dialog box (1802) for adding new entries. It contains the following fields and controls:

- ユーザ名 (User Name): Text input field (1831)
- プロセス定義名 (Process Definition Name): Text input field
- 契約形態 (Contract Type): Radio buttons for 共有 (Shared) and 固定 (Fixed)
- 優先度 (Priority): Text input field
- 実行多重度 (Execution Multiplicity): Text input field
- OK button

【図 1 9】

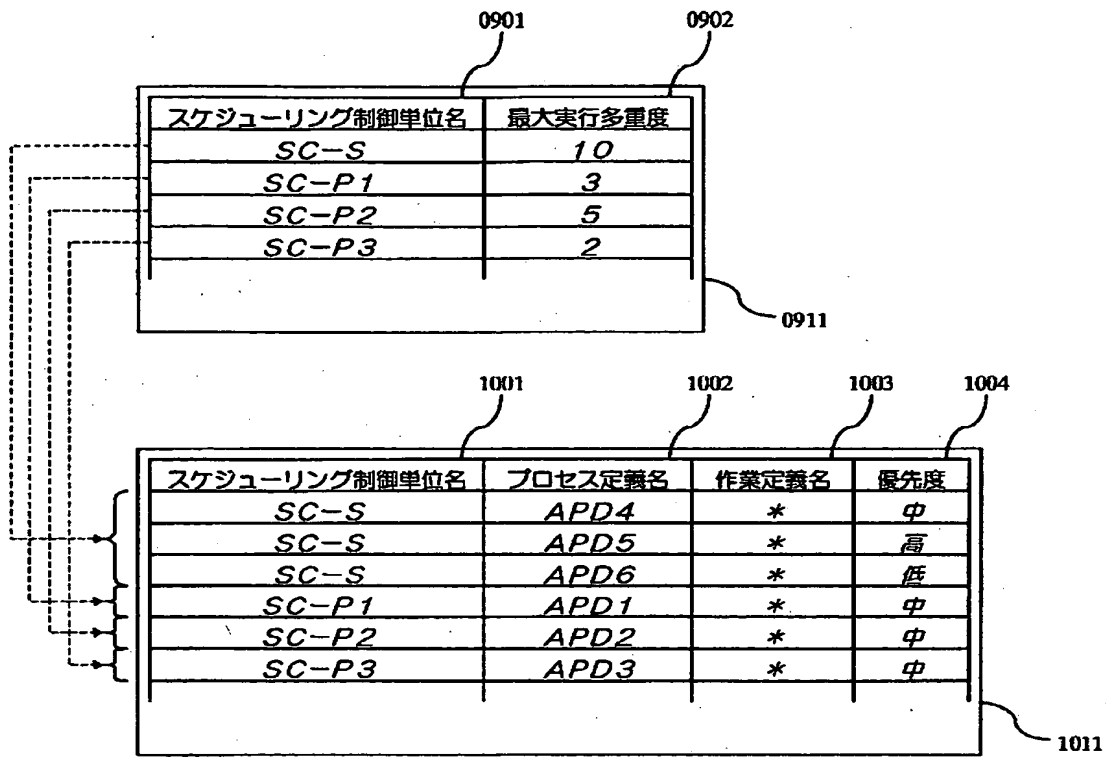
図 1 9

ユーザ名	プロセス定義名	契約形態	優先度	実行多重度
USR1	APD1	固定	中	3
USR1	APD2	固定	中	5
USR2	APD3	固定	中	2
USR3	APD4	共有	中	10
USR4	APD5	共有	高	10
USR5	APD6	共有	低	10

1901

【図20】

図20



【図21】

図21

業務プログラム情報	メモリ使用量 (計算機リソース1)	ディスク使用量 (計算機リソース2)
Program1	1 MB	10 MB
Program2	10 MB	100 MB
Program3	50 MB	50 MB
Program4	30 MB	200 MB

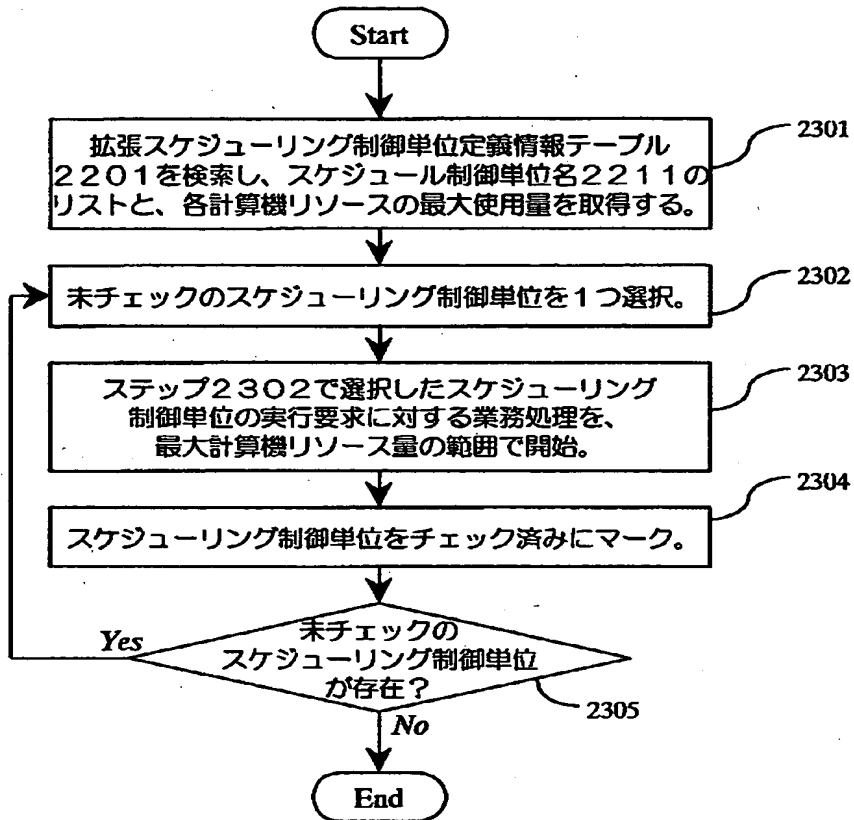
【図 2 2】

図 2 2

スケジューリング制御単位名	最大メモリ使用量 (計算機リソース1)	最大ディスク使用量 (計算機リソース2)
SC1	1 GB	4GB
SC2	500 MB	2GB
SC3	200 MB	1GB

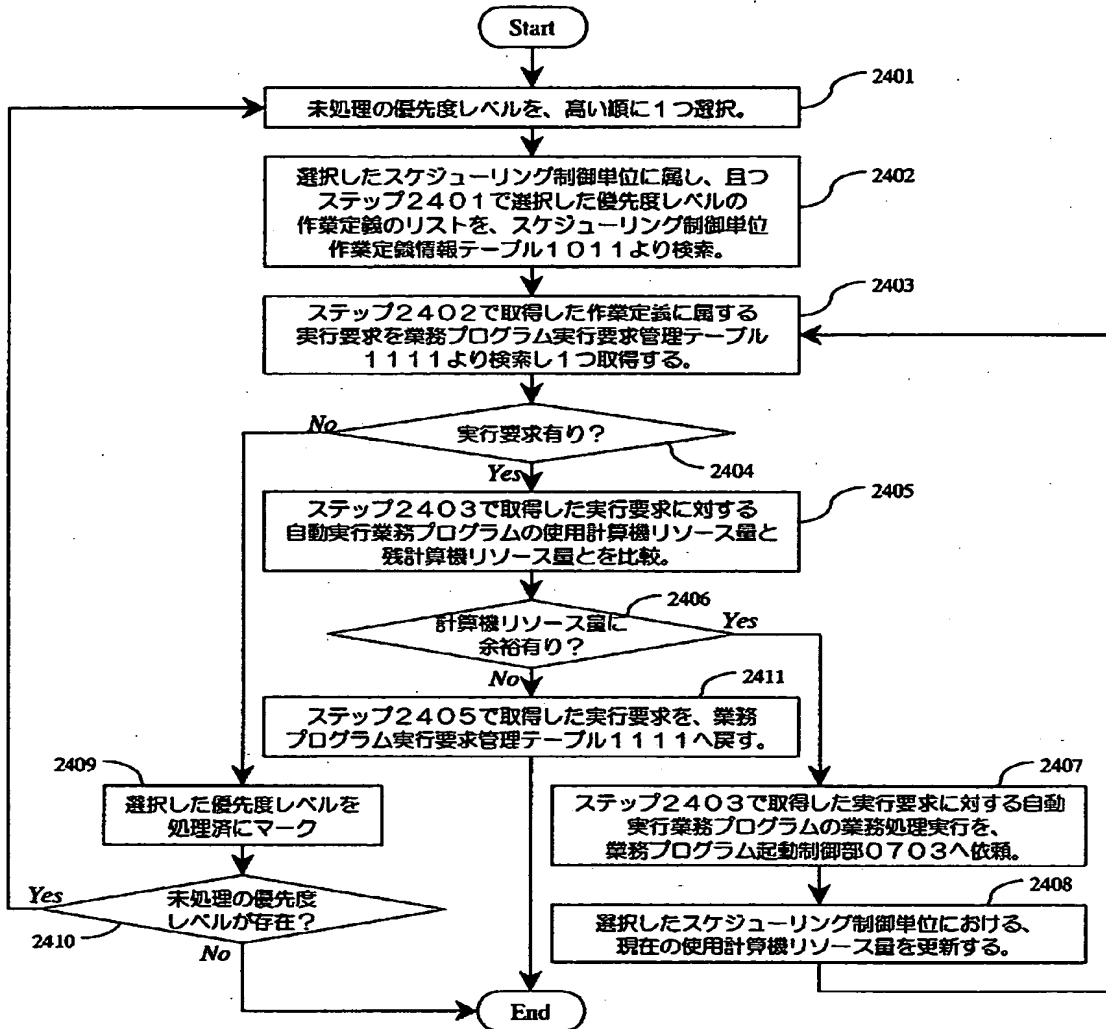
【図 2 3】

図 2 3



【図 24】

図 24



【書類名】 要約書

【要約】

【課題】

作業に対する処理を実行する業務プログラムが使用する計算機リソースの配分を考慮したワークフロー管理を実現できるようにする。

【解決手段】

業務における複数の作業と該作業間の処理順序とを定義するプロセス定義情報を保持し、該プロセス定義情報に基づいて前記作業の実行管理を行うワークフロー管理方法において、複数の前記作業の識別情報と、該作業の処理を行う業務プログラムの実行可能数とが対応付けられて格納されている制御情報を保持し、

前記作業の処理を行う業務プログラムを実行する必要がある場合、既に実行している業務プログラムが使用している計算機リソースが前記実行する必要がある作業に対応して格納されている所定値以下であるか判定し、所定値以下の場合は上記実行する必要がある業務プログラムを実行する

【選択図】 図 8

出 願 人 履 歴 情 報

識別番号 [000005108]

1. 変更年月日 1990年 8月31日
[変更理由] 新規登録
住 所 東京都千代田区神田駿河台4丁目6番地
氏 名 株式会社日立製作所